Evaluating the Effectiveness, Generalizability, and Explainability of Video Swin Transformers on Automated Pain Detection

Thesis M.Sc. Artificial Intelligence

Maximilian Rau
m.rau@students.uu.nl

2024

1st supervisor: Assist. Prof. Dr. I. Önal Ertuğrul
2nd supervisor: Prof. Dr. Albert Gatt

Department of Information and Computing Sciences
Faculty of Science

Universiteit Utrecht

Contents

Abstract

Recent advancements in computer vision, particularly with transformer-based models, offer promising potential for establishing new benchmarks in automated pain assessment through facial expressions. This thesis explores the efficacy of the Video Swin Transformer (VST), a recent approach that leverages temporal dynamics and offers a potential for nuanced detection capabilities of pain through varying scales. Our study involves applying the VST and comparing its performance against other transformer-based state-of-the-art models such as the Swin Transformer and the Vision Transformer (ViT). Through ablation studies, we demonstrated the positive impact of incorporating a higher temporal depth length into the model. Additionally, we evaluated the use of Focal loss to mitigate the issue of an imbalanced class distribution found in the UNBC McMaster dataset, which turned out to be insufficient. Furthermore, our research also focused on the generalizability of our models across different datasets, highlighting the need for more diverse datasets in training phases. Through the extraction of attention maps, we gained insights into the explainability, particularly the focus points of our models, confirming their utilization of pain-related regions for decision-making. The results were promising: our best models, VST-0 and VST-1-TD, set new benchmarks with F1-scores of 0.56 ± 0.06 and 0.59 ± 0.04, respectively, and achieved comparable state-of-the-art AUC scores of 0.85 ± 0.04 and 0.87 ± 0.03. This thesis underscores the potential of the VST architecture not only in automated pain assessment but also its broader applicability in the analysis of facial expressions.

1 Introduction

Pain, defined by Merskey et al. (25) as "an unpleasant sensory and emotional experience associated with actual or potential tissue damage, or described in terms of such damage", is critical in clinical diagnostics and treatment. Effective pain assessment plays an essential role in healthcare, that can guide treatment decisions and patient management. However, inaccurate pain assessment can lead to serious consequences, including inappropriate treatment.

In recent years, automated pain assessment through tracking facial expressions has gained more interest. Facial expressions, especially those specific to pain, have emerged as reliable indicators of pain experience. Unlike other modalities, such as neural activity monitoring, facial expressions can be conveniently tracked through video processing without the need for a complex setup. While expert assessment of pain via facial expressions is possible, it is often costly and impractical for frequent and real-world applications. Therefore, automation presents a promising solution for continuous and accurate monitoring of pain experiences over extended periods. Using advanced technologies to objectively measure and interpret facial pain indicators has the potential to improve patient outcomes in clinical settings.

The field of computer vision is growing quickly, with a clear shift from traditional Convolutional Neural Network (CNN) approaches to models based on transformers. Vision Transformers (ViTs) (20) have set new state-of-the-art performance in various vision tasks, including pain detection and pain intensity estimation. However, ViTs are not without limitations, such as their high computational complexity. To address these issues, new architectures like the Shifted Window (Swin) Transformer (47) have been developed. Swin Transformers have gained popularity due to their ability to outperform ViTs, particularly in frame-level image processing. However, in the context of pain, where facial dynamics play an important role, the capabilities of the spatiotemporal counterpart, the Video Swin Transformer (VST) (48), are particularly promising.

This research thesis aims to contribute to the field of automated pain assessment by investigating the impact of the Video Swin Transformer on pain detection. With the inclusion of spatiotemporal information, the VST is a potentially better alternative to the original Swin Transformer for automated pain assessment. We hypothesized that the architectural advancements of the VST over the ViT would also positively influence the pain detection performance. Additionally, this study further investigates the generalization capabilities of transformer-based models across different pain contexts

and their explainability, providing a holistic understanding of transformer-based models in automated pain assessment. As this research is centered around the VST, further ablation studies about design and training decisions are examined.

The structure of this thesis is as follows: it begins with the defined research questions in Chapter 2, followed by a detailed literature review divided into two parts. The first part (Chapter 3) provides a general overview of pain and its assessments, while the second part (Chapter 4) discusses the relevant literature and technical foundations crucial for our methodology. This is followed by a description of the used datasets in Chapter 5. Subsequently, in Chapters 6 and 7 an overview of our methods, including the model pipeline and experiments, is presented. Finally, the obtained results are presented, followed by a discussion and conclusion in Chapters 9 and 10.

2 Research Questions

The project's preliminary main and sub-research questions are outlined in this chapter. The subsequent literature study discusses the basis and motivation, while the methodology and experiment section provide the used methods and the planned experiments to address these research questions. The following research questions are considered in the scope of this study:

- **Main Research Question:** How do Video Swin Transformers perform in the automated assessment of pain through facial expressions?

To investigate the performance of the Video Swin Transformer model on this specific task, we conducted several experiments using a variety of evaluation methods. The study also involves comparative analysis with other state-of-the-art models like (1) the original Swin Transformer and (2) the Vision Transformer, along with results from prior research on the same task. Additionally, the research project covers multiple ablation studies, including the analysis of the generalizability and explainability of these models in automated pain detection. The following sub-research questions address more specific aspects within the study's broader scope:

- **1. Sub-research Question:** How does incorporating temporal dynamics of pain at the video-level impact the performance of automated pain detection compared to solely frame-level analysis?

The first sub-research question is addressed by the comparison of the Video Swin Transformer (video-level) with its spatial counterpart, the Swin

Transformer (frame-level). With this comparison, we can analyze how including temporal information has an impact on automated pain detection performance.

- **2. Sub-research Question:** To what extent does increasing the temporal depth input of Video Swin Transformers enhance pain detection capabilities?

This sub-research question explores if extended temporal depth, which captures more information about pain dynamics, can improve performance. To answer this question, the study evaluates a Video Swin Transformer with an extended temporal depth and compares it with the same model without the extension.

- **3. Sub-research Question:** How does the use of Focal loss during training on the imbalanced UNBC McMaster dataset, in comparison with oversampling techniques, impact the detection of pain?

This sub-research question explores strategies for handling dataset imbalances, specifically with the UNBC McMaster dataset, known for its skew in pain class distribution. To address this, the study compares the Video Swin Transformer model trained on UNBC McMaster with two different approaches: (1) oversampling the minority pain class, a common technique in the literature, and (2) Focal loss, another method designed to deal with class imbalances during training. This comparison aims to evaluate the effectiveness of Focal loss and which approach achieves better pain detection performance.

- **4. Sub-research Question:** How do Video Swin Transformer-based pain detectors generalize across different pain contexts?

To tackle this sub-research question, we are conducting experiments using two well-known and widespread pain datasets: (1) the UNBC McMaster, which focuses on shoulder pain, and (2) BioVid, which centers on heat pain. By training the model on the UNBC McMaster dataset and afterward testing it on the BioVid dataset, we aim to get insights into the Video Swin Transformer's capability to generalize across these pain contexts. The generalizability results of the Video Swin Transformer are also compared to those of other state-of-the-art models to evaluate their relative performance in cross-dataset validation.

- **5. Sub-research Question:** Can model-specific explainability methods generate plausible explanations for the outputs of Video Swin, Swin, Vision Transformer-based pain expression detection models, and how do the explanations generated differ among the model architectures?

To address this question, we are applying model-specific explainability techniques to explain the decisions made by these models in detecting pain expressions. Afterwards, the generated explanations are analyzed, compared, and evaluated for plausibility. Through this research, we try to overcome the gap between complex model outputs and meaningful, interpretable findings in the domain of automated pain expression detection.

3 Literature Study: Pain Assessment

In this part of the literature study, a general overview of the pain phenomena and their assessments is given. First, pain and its responses are analyzed from the biological point of view, with a focus on behavioral responses in the form of facial expressions. In this section, the Facial Action Coding System (FACS), facial expressions associated with pain, and challenges recognizing these specific facial expressions are discussed. Secondly, clinical pain assessment methods, which can be categorized into self-report and observable techniques, are described, and their benefits and drawbacks are pointed out, motivating the need for automation in this field. Lastly, automated pain assessment is examined, showing the problem definition of this task, its variability in approaches, and the most important challenges. Therefore, several studies in this domain are inspected, evaluated, and classified according to the topic.

3.1 Pain responses and facial expression

As mentioned in the introduction, pain is a complex human experience on a sensory and emotional level. In the first place, its purpose is through psychological and physiological elements to warn and protect living beings from actual or potential damages (66). This protective mechanism encourage the affected one to act appropriately to minimize the harm. For instance, if someone puts their hand on a hot stove, this individual will experience pain throughout the heat and, hence, has a reflexive withdrawal of the hand from the stove. In this case, with pain, the body gives a signal to act and avoid further damage like severe burns. In general, pain can occur in several forms

(e.g., acute, chronic, or neuropathic pain), as well its causes can differ, for instance, by injuries, illnesses, or inflammations (56).

Despite these differences, the human body's response to pain is often similar, whereby the responses to pain can be mainly divided into two categories: physiological and behavioral. Physiological responses are understood to be changes in neural activity, general vital signs, and hormonal release (10), while behavioral responses refer to the observable actions and expressions - more specifically facial expressions, body movements, and vocalization - that individuals show in response to pain. With behavioral responses, an individual communicates the pain and tries to get potential help, likely due to evolutional reasons to increase the survival rate (90). Particularly facial expressions function as a reliable indicator of pain as there are facial expressions specifically associated with pain, which are also relatively consistent over a whole variety of clinical pain situations (65) (75) (64). Studies such as the work by Simon et al. (76) showed that there is a significant distinction between facial expressions that are pain-related and those that represent other basic emotions. Furthermore, Kunz et al. (43) manifests the increase of facial movements with rising pain intensity, pointing out the distinctiveness among pain intensities.

Research related to facial expressions or facial recognition tasks is usually based on the Facial Action Coding System (FACS) released by Ekman et al. (22). It is considered the gold standard for objectively analyzing facial expressions. More specifically, it describes facial expressions based on Action Units (AUs) each representing different facial muscle activity and breaking down facial expressions into unique elements of muscle movements. In total, the coding system contains 44 AUs, by which a subset is related to pain. This subset includes the following AUs (65): Lowering of brows (AU4), cheeks raising (AU6), lid tightening (AU7), nose wrinkling (AU9), rising the upper lip (AU10), and eye closing (AU43).

An example of a painful facial expression with AU6, AU9, AU43 is illustrated in Figure 1. Important to note is that these AUs do not imply the existence of a single, uniform facial expression of pain that remains constant across all individuals and situations. Instead, individuals frequently show partial components of this subset or mix these specific facial actions in varying ways (80).

As a metric for the quantification of pain based on these facial expressions, the study by Prkachin and Solomon (65) introduced the Prkachin-Solomon-Pain-Intensity (PSPI) scale. This scale, well-known in the domain of facial pain research, offers a standardized methodology to quantify pain intensity by evaluating specific pain-related facial AUs. In the PSPI scale,

Figure 1. *Example of pain-related Action Units - AU6, AU9, and AU43 (89)*

each action unit is rated on a six-point ordinal scale, where 0 denotes absence and 5 represents maximum intensity. The complete formula to calculate the PSPI is shown below:

$$\text{PSPI score} = AU4 + \max(AU6, AU7) + \max(AU9, AU10) + AU43 \quad (1)$$

In this formula, AU43 is binary, taking values of either 0 or 1, whereas the other action units (AU4, AU6, AU7, AU9, and AU10) can reach up to an intensity level of 5. Consequently, the resulting pain scale covers 16 distinct levels (3×5+1=16). Overall, the scale is an important metric in observable pain assessment methods and is used in annotating pain intensities within datasets of facial expressions, for example in the UNBC McMaster pain dataset (49).

Although specific facial expressions are indicative of pain (76), recognizing them is not always straightforward and provides challenges. One of the difficulties is that pain often appears together with other emotions, leading to ambiguous or blended facial expressions (90). Additionally, individual variation makes this challenge even more difficult (53). People differ significantly in their facial expressiveness, with some showing noticeable pain signs and others possibly expressing less obvious, more difficult ones. These differences in expressiveness can make consistent and accurate pain recognition a challenging task.

3.2 Clinical pain assessment

Pain assessment is an essential aspect of clinical care, as it plays a central role in diagnosing and managing several medical conditions. The evaluation of pain often relies on the individual's self-report, where they consciously communicate their perception of pain (16). This self-reporting can take various forms, including spoken or written communication and even gestures. In clinical settings, self-reporting methods are typically categorized into three main types: Numerical Rating Scales (NRS) (21), Visual Analogue Scales (VAS) (54), and Verbal Rating Scales (VRS) (18).

- Numerical Rating Scales (21): This method involves using a numerical scale to assess pain intensity. Patients are asked to rate their pain on a scale, which is useful for maintaining pain diaries and tracking changes over time.
- Visual Analogue Scales (VAS) (54): VAS utilizes a visual line with two endpoints, representing "no pain" and "extreme pain." This scale offers a good level of differentiation in pain intensity and is often favored for its simplicity.
- Verbal Rating Scales (VRS) (18): VRS assesses pain through various verbal descriptors, allowing patients to choose words that best describe their pain. This method is particularly useful for patients who may find it challenging to quantify pain numerically.

Additionally, there are other pain assessment scales tailored to specific patient groups, such as the elderly (96). The choice of pain assessment method depends on the clinical environment and the patient's condition. Self-report methods are often considered the gold standard for pain assessment because pain is a highly subjective experience (16). Subjective assessments align well with self-reporting, as they offer insights into the individual's unique pain experience. Moreover, self-reporting is convenient to apply and is economically advantageous in clinical practice. However, it is essential to acknowledge that self-report methods are not always feasible or suitable for all patients. Self-report methods can be limited by the need for verbal communication and cognitive functionality, which may be impaired in certain patient populations (33). Moreover, these methods can introduce bias and variance due to their goal-oriented and controlled nature (15).

As an alternative to self-reporting, observational pain assessment scales are available. These scales rely on the observation of behavioral pain responses, such as facial expressions, movements, or vocalizations, to assess

pain. Some observational scales also incorporate physiological responses. Observational scales are often designed for specific patient populations, including infants and pre-verbal toddlers, elderly individuals with severe dementia, and critically ill or unconscious patients (33)(96). Facial expression analysis, often using the FACS, is a prevalent method in these observational scales. They can be invaluable for patients who cannot provide self-report assessments. Indeed, they also have their drawbacks. The primary disadvantage is the effort required for training and experience to apply them accurately (23)(37)(2). In clinical environments facing worker shortages and economic pressures, the use of observational scales can be challenging to implement consistently, leading to potential inaccuracies in pain assessment.

3.3 Automated pain assessment

The research in (30) showed that frequent and accurate pain monitoring is essential for the patient's outcome as it can significantly improve the patient's diagnosis and, consequently, lead to better treatment for each individual. In the previous chapter, we highlighted the limitations of observable pain scales, which often result in inaccurate and infrequent pain measurements. Addressing this issue, automation presents a promising solution for more effective pain assessment. In this chapter, we will explore the general pipeline of Automated Pain Assessment (APA) systems, along with an overview of previous research and design decisions.

3.3.1 Pipeline of Automated Pain Assessment system

Figure 2. *Pipeline APA system*

The general pipeline of an APA system includes several steps, as depicted in Figure 2. To begin, a signal input is required to assess pain, providing information about the pain status of the individual in question. Various modalities, including single-modal and multimodal approaches, have been explored in previous research. For modalities in pain assessment, physiological and behavioral responses can be considered, as discussed in Chapter 3.1. Behavioral responses, which can often be recorded in a contactless and non-intrusive manner, offer a practical advantage over physiologi-

cal responses. Some physiological responses, such as brain activation data (e.g., EEG recordings), are generally limited to experimental settings, expensive, and require significant preparation. Hence, behavioral responses are frequently preferred for signal acquisition in many investigations. Notably, facial expressions have proven to be a reliable indicator of pain (65). Consequently, most of the previous approaches in APA have employed camera-based methods to analyze pain through facial expressions (89). Due to the practicality and performance of facial expressions as a modality for pain assessment, our research also centers around this modality.

In the second step of the pipeline, the input modalities often require preprocessing before they can be used in a classifier model. This preprocessing may involve feature extraction, label preprocessing, face frontalization, or data cleaning. Feature extraction can be often seen as a separate step in the pipeline, as it includes often separate methods. For simpler machine learning classifiers such as Support Vector Machines (SVM) or Logistic Regression (LR), feature extraction from images is a common requirement. Various feature extraction methods, including Histogram of Oriented Gradients (HOG) or generic appearance features, are applied to create features that can then be used in these classifiers. Conversely, for approaches utilizing deep neural networks, feature extraction as a preprocessing step is often not required, as these networks can automatically learn and extract relevant features from the raw data.

In the classification step, there is a distinction between conventional machine learning approaches and deep learning methodologies. We follow this categorization as we go through representative works in the field in the next subchapter. Among the various deep learning methodologies, attentive models can be seen as a subgroup. The Video Swin Transformer, for instance, falls under this category.

Another layer of granularity in classifier methodologies is based on their input dimensions. The input dimensions can generally be categorized into two groups: those centered on spatial aspects, which operate solely on frame-level data, and those involving spatiotemporal methods, incorporating a temporal dimension. The latter is especially crucial in contexts like pain assessment, where the progression and transition of facial expressions or other indicators over time provide valuable insights. Lastly, the output phase of APA systems can be typically distinguished in binary detection and pain intensity estimation.

In the following subchapter, we will discuss representative works in the field of automated pain assessment, providing descriptions and analyses of their methodologies. Note that we consider in the following chapter

only related work working with non-attention-based approaches as attentive approaches are discussed in the Chapter 4.

3.3.2 Representative work of non-attention-based automated pain assessment

Non-Deep Learning Approaches. Initial developments in the field of automated pain assessment were based on pattern matching and traditional machine learning techniques. Several studies investigated different feature extraction methods, and these preprocessed representations were employed as inputs in subsequent machine learning models instead of raw pixels. In their first efforts at automating pain assessment, Ashraf et al. (5) aimed at frame-level pain detection, recognizing the presence or absence of pain using the UNBC McMaster shoulder pain dataset. In order to capture facial appearances and shapes as features from the images, they used Active Appearance Models (AAM). In the final step, the extracted representations derived from the AAM were then, with a recall of 82% and a false-positive-rate of 30%, classified into pain or no-pain categories using SVM. Another successful study in the initial investigations was conducted by Khan et al. (39). This research study proposed an alternative feature extraction approach, again using the UNBC McMaster dataset. They employed the Pyramid Histogram of Orientation Gradients (PHOG) to capture shape information, while appearance information was extracted using the Pyramid Local Binary Pattern (PLBP). This combination is intended to yield a more discriminative representation of facial expressions. With these extracted features together, the study experimented with several machine learning models, including SVM, Random Forest, Decision Tree, and 2-Nearest Neighbors (2NN). The results indicated promising performance, particularly when employing the 2NN model (96.9% accuracy). Furthermore, some works also applied Scale Invariant Feature Transform (SIFT) to extract feature vectors. For instance, the work by Neshov and Manolova (60), which focused on facial expression analysis for automatic pain recognition with the same dataset as the previous studies. The first step involved locating specific landmarks on the face using the Supervised Descent Method (SDM). Subsequently, feature vectors were extracted employing the SIFT, followed by an SVM classifier. This study showed that using SIFT achieved an accuracy of $\sim$96%, and is comparable to the best-performing model mentioned by Khan et al. (39).

One of the limitations of the aforementioned studies was their exclusive focus on spatial information. These do not take the potential significance of the temporal axis under consideration. As mentioned previously, facial expressions, especially those relating to pain, can be best understood within a

temporal context as dynamics of facial expressions often appear across multiple frames. Recognizing this potential, Werner et al. (87) took a different approach by focusing on the temporal dynamics of facial expressions. Instead of just relying on frame-level features, they employed "activity descriptors", which are essentially sequence-level feature signals derived from facial landmarks and head pose. Their experiments were not limited to the UNBC McMaster database but extended to the BioVid Heat Pain Database as well. Their temporal integration not only showed the importance of sequence-based information but also outperformed the previous state-of-the-art methods in pain classification on both databases. Furthermore, another approach to use the temporal dimension was developed by Werner et al. (88). This research centered on spatiotemporal pain detection using the BioVid Data Set. Particularly, they employed optical flow analysis on a frame-by-frame basis, allowing them to extract spatiotemporal features.

Both papers highlight the benefits of temporal integration in the field of automated pain detection. Their developments provide a basis for our experiments, in which we are using the Video Swin Transformer model to further explore temporal dynamics.

Deep Learning Approaches. With the advancement of deep learning, researchers explored many novel techniques in APA systems. While some studies applied pure Deep Neural Networks (DNNs), others (92)(72)(44) used these networks to produce high-level representations. Often in combination with traditional handcrafted features, these representations were then fed into another deep neural network or traditional machine learning models for pain prediction. This idea of combining both handcrafted and deep-learned features was presented in the research by Yang et al. (92), where the authors showed a fusion method that merges low-level local descriptors with high-level neural network-produced features.

Research conducted by Semwal and Londhe (71) introduces a computationally efficient convolutional neural network (CNN) for pain recognition. Unlike existing techniques that rely on handcrafted features or deep, computationally expensive CNNs, this paper proposes a shallower CNN with only three convolutional layers. Evaluated on the UNBC McMaster shoulder pain dataset, the approach achieved 93.34% accuracy in multiclass pain recognition, outperforming handcrafted feature-based methods and other deeper CNNs in performance.

The study by Tavakolian and Hadid (78) presents a spatiotemporal convolutional network designed for pain intensity estimation from facial video

sequences. The presented network captures various temporal facial expression variations using 3D convolutions with differing temporal depths. Experimental results on the UNBC-McMaster Shoulder Pain and BioVid datasets demonstrate the strength of including temporal information. Notably, in their ablation studies, they found out that their model with a temporal depth of 32 outperformed the ones with a lower number. Investigations further into the temporal aspect were made by Rodriguez et al. (67), highlighting the effectiveness of Long short-term memories (LSTM) in pain detection. In this study, a two-step model was applied starting with a CNN model capturing first the spatial information. Following this, an LSTM model processed the CNN's output, covering the spatiotemporal relationships in the image sequences with a temporal depth of 16 frames. It is interesting to note that although the CNN performed admirably on its own, the addition of the LSTM model increased the area under the curve (AUC) measure by about 4%. This shows again the potential of spatiotemporal information in the field of automated pain assessment.

Other representative work focusing on attentive transformer models, like Vision Transformers, will be elaborated on in the following chapter.

4 Literature Study: Methodology

The relevant studies regarding our methodology are covered in detail in this section of the literature study. Since the Vision Transformer and the Video Vision Transformer (ViViT) are the predecessors of the Swin transformer family and share similar principles and characteristics, it is necessary to understand their features and capabilities. Following the review of the ViT design and its associated work with automated pain assessment, we will continue to emphasize the original Swin Transformer architecture and its benefits over the ViT. This will then form the basis of an analysis of the Video Swin Transformer. Following, the theoretical background and associated research on cross-dataset validation within this field will be explored in order to address the cross-dataset validation in our experiments. Additionally, interpretability and explainability are discussed, together with model-specific explainability methodologies, which provide the framework for the fifth subresearch question. Last but not least, we will talk about imbalanced class distribution in datasets, which is common in the facial expression domain, and how to handle this problem.

4.1 Vision Transformer and Video Vision Transformer

Natural language processing (NLP) had a revolution with the introduction of transformer architectures, which were introduced in the paper by Vaswani et al. (81). However, the first time the power of transformers was successfully used in the field of computer vision was with the Vision Transformer (ViT) (20). Its performance was shown to not only be comparable but often outperform the previous state-of-the-art architectures, which were primarily based on CNNs, or Recurrent Neural Networks (RNNs), in a variety of computer vision tasks. The advantage of ViT is, unlike CNNs and RNNs, its capability to process global characteristics of images using self-attention mechanisms, which more efficiently capture long-range relationships in data. In the next subchapter, we will give a technical overview of the ViT, including its spatiotemporal counterpart, the ViViT.

4.1.1 Technical overview

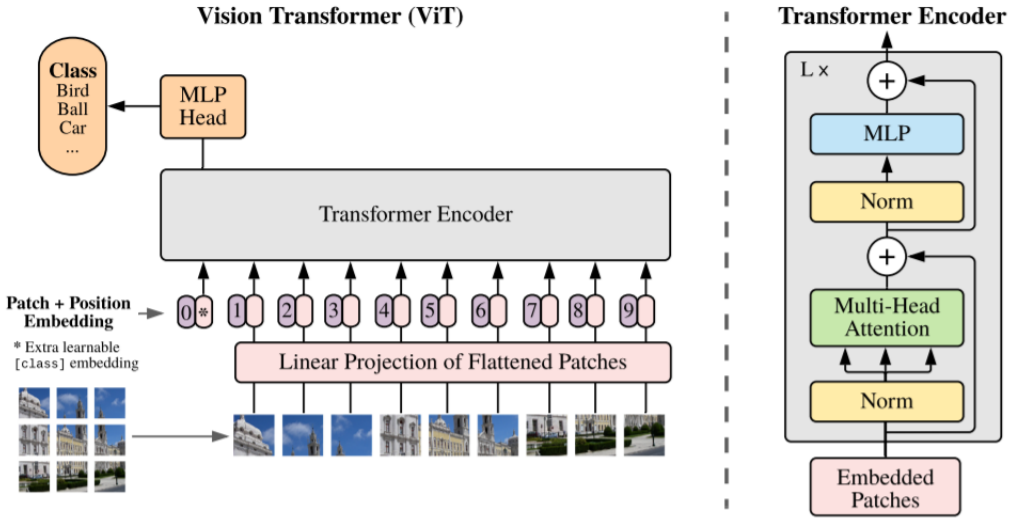


Figure 3. *Overview of the ViT architecture (20)*

The ViT is an adapted version of the transformers used in NLP to be used for image data. An overview of the ViT architecture is illustrated in Figure 3. The entire architecture is displayed on the left, while a more detailed cutout of the transformer encoder block is displayed on the right. As can be observed, the whole architecture consists of several steps, which are explained as follows:

Image patch division, linear projection, and embeddings. In the context of NLP, transformers work on a sequence of words. In order to apply this idea to computer vision, an image is processed by dividing it into fixed-sized patches throughout. The patch division is required because, when combined with the transformer design, which operates globally, treating every pixel as a patch would cause a computational overload. In the second stage, these patches are flattened into a single vector termed patch embeddings, which are obtained by concatenating the channels of all the pixels in a patch and then linearly projecting it to the required input dimension. This series of embeddings is preceded by an embedding of a learnable class. Moreover, positional embeddings are utilized to represent local context because self-attention is computed globally inside transformers. This set of embeddings, which now includes both the content and the spatial information of the image, is given into the transformer encoder.

Transformer block. The patch embeddings will be passed into a transformer encoder block as input in the next stage. Notably, there may be many following transformer encoders across the entire ViT architecture. One encoder block (see right side of Figure 3) consists of a multi-headed self-attention (MSA) and a multi-layer perceptron (MLP) block which introduces two layers with Gaussian Error Linear Unit (GELU) non-linearity. Additionally, each MSA and MLP block has a normalization layer in front of it, which improves the efficiency of deep encoding transformer learning (85). A residual connection follows each of these blocks, adding the layer's input and output (7).

Since the **scaled dot-product attention** (81) (see left side of Figure 4) is a component of the multi-headed attention block, it is crucial to understand the idea of self-attention before moving on to it. The self-attention can be calculated using the Equation 2:

$$Attention(Q, K, V) = softmax(\frac{QK^T}{\sqrt{d_k}})V \tag{2}$$

The attention mechanism makes use of three variables: Query (Q), Key (K), and Value (V), as can be seen in the formula. Each of these three variables —which are matrices— was calculated from an input vector and three corresponding trained weight matrices. The dot product of the Query and Key matrices, scaled by the root of their length $\sqrt{d_k}$, and multiplied by the Value matrix are used to calculate the self-attention's final output.

Returning to the **multi-headed attention** block, its structure is depicted on the right side of Figure 4. It uses multiple self-attention layers parallel, each containing a set of training matrices. The self-attention layers'

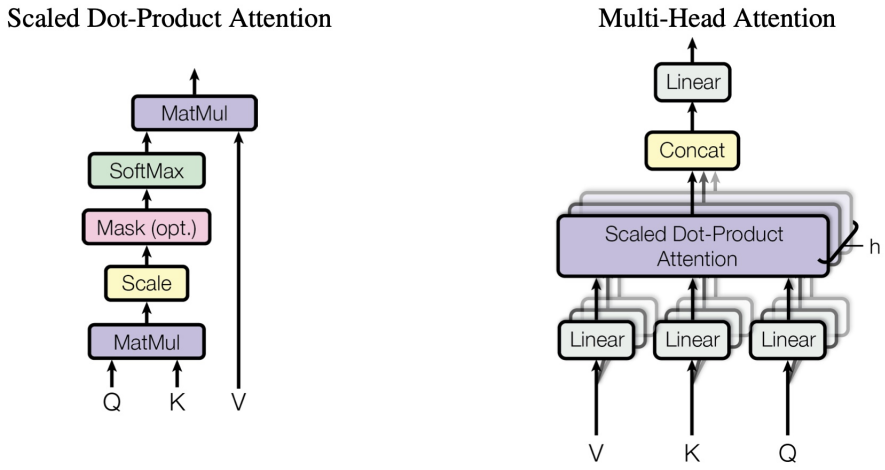


Figure 4. *Scaled Dot-Product Attention (l) and Multi-Head Attention (r) (81)*

outputs are concatenated, and the resulting matrix is multiplied by another training matrix to produce the output. The underlying idea of multi-head attention is to enable varied attention to different parts of the sequence with each pass.

Classification head. Lastly, a classification head is attached to the transformer encoder's output. During pre-training, an MLP with a single hidden layer, and during fine-tuning, a single linear layer construct the classification head. It is required for mapping the learned feature representations to task-specific outputs, such as class labels in image classification tasks.

Video Vision Transformer. To be able to handle spatiotemporal data like sequences, which include both a time dimension and spatial dimensions, the Video Vision Transformer was developed as an adaptation of the ViT. The ViViT was published by Arnab et al. (4) and works similarly to the ViT along with some adaptations in the embeddings and the transformer encoder. Firstly, regarding the authors, embedding spatiotemporal data can be approached via Uniform Frame Sampling or Tubelet Embedding. By independently embedding uniformly sampled frames, the former concatenates these tokens together to create a large 2D image, while the latter expands embedding to 3D by linearly projecting spatiotemporal "tubes," which are similar to 3D convolution. Secondly, the research paper describes four spatiotemporal model variations: (1) Spatio-temporal attention model which processes all spatiotemporal tokens together (2) Factorised encoder model

that employs separate transformer encoders for spatial and temporal interactions (3) Factorised self-attention (4) Factorised dot-product. The four approaches can be observed in Figure 5. It was found that the spatio-temporal attention model is the most straightforward approach and outperformed the others, although it needs to be considered that the computational complexity is quadratic.

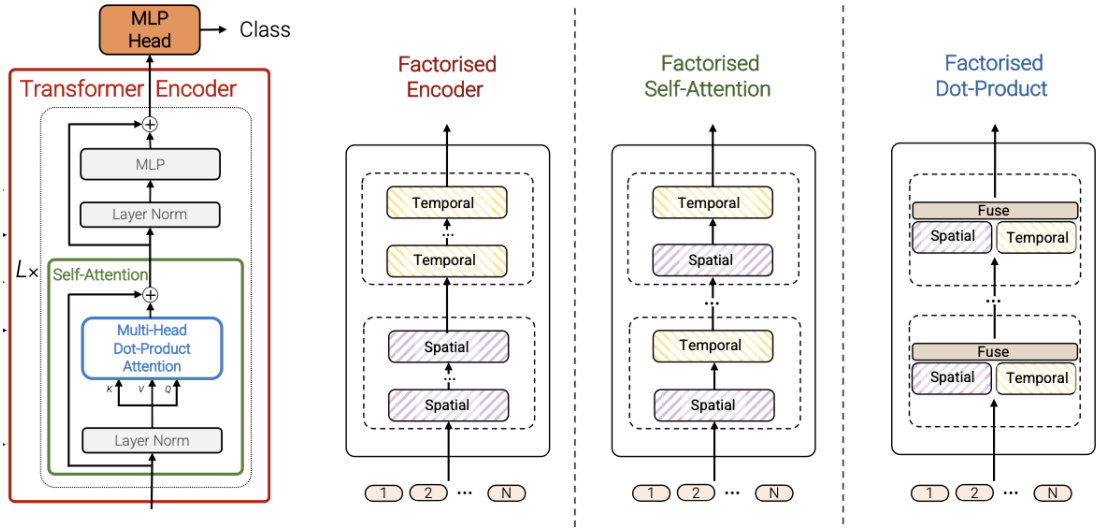


Figure 5. *Spatiotemporal model design variations for ViVit (4)*

4.1.2 Related work using ViT and ViViT

Previous research on the ViT and the ViViT showed the potential of this architecture and the used attention mechanism. Starting with the foundational paper of the ViT (20), the model marked a shift in image recognition tasks by achieving state-of-the-art results on benchmarks like ImageNet with 88.55% accuracy and CIFAR-100 with 94.55%. Similarly, the original paper of the ViViT model (4) has shown competitive performance on large-scale video datasets such as Kinetics 600 with a Top-1 accuracy of 85.8%. Both, the ViT and its spatiotemporal counterpart, show the potential to outperform previous non-attentive architectures like ResNet or EfficientNet.

Facial expression recognition and AU detection tasks are related to automated facial pain assessment. Moreover, the application of the ViT and ViViT models on these tasks has been as well explored with promising outcomes. The approach by Sun et al. (77) uses ViTs for part-based face recognition. They introduce a so-called part fViT pipeline, using a lightweight

CNN to predict facial landmarks, and then apply ViT on patches around these landmarks for enhanced face recognition. This method achieves state-of-the-art accuracy on multiple benchmarks, for instance on CFP-FP with an accuracy performance of 99.21%. Similarly, but on a spatiotemporal level, the study by Tu et al. (82) demonstrated the effectiveness of ViViTs in AU detection, introducing an approach that integrates the Video Vision Transformer with a CNN backbone to efficiently capture temporal facial changes. This hybrid model not only significantly outperformed the baseline model of the Affective Behavior Analysis in-the-wild (ABAW) competition in 2023 by a notable 14% but also showed comparable results with the top-performing teams from the previous year's competition. Another approach emphasizing the aspect of multiple scales for facial expressions, which is also an important characteristic of our used Video Swin Transformer, was introduced through the Progressive Multi-Scale Vision Transformer (PMVT) by Wang et al. (84). The proposed architecture, which is at its core based on the ViT, includes a multi-scale self-attention mechanism that can flexibly attend to a sequence of image patches to encode the important features for AUs. Experimental results show that PMVT improves the accuracy of several AUs on the BP4D and DISFA datasets.

For the direct application of ViT in pain assessment, the research by Xu and Liu (91) was one of the first focusing on pain estimation from facial expressions using transformer-based architecture. The reported approach in this work focuses on end-to-end pain intensity estimation on the spatiotemporal level and consists, similar to the mentioned studies on AU detection (77) (82), of both a CNN and a transformer. Before being processed by a transformer model that predicts the pain intensity, pain-related features are first recognized and retrieved from the input images using a ResNet architecture with bottleneck attention modules. One of their findings was that a pure transformer alone does not work for pain assessment. In further developments, Fiorentini et al. (27) have challenged the findings by Xu and Liu (91) regarding the ineffectiveness of pure transformers in pain assessment. Introducing a fully attentive pipeline using ViT on spatial and ViViT spatiotemporal levels, their work has set new benchmarks in the domain. Trained on the 3D-registered and frontally-aligned UNBC-McMaster dataset, their best models demonstrated state-of-the-art performance in binary pain detection. Both ViT and ViVit could achieve F1-scores of 0.55. Interestingly, their ViViT, which incorporates temporal information through uniform frame sampling, did not outperform its spatial counterpart. This suggests that the spatiotemporal approach in transformer models may not be more effective than the spatial aspect alone.

4.2 Swin Transformer

Even though the ViT is highly effective at several computer vision tasks, it has drawbacks. ViTs, for example, have difficulty processing images with a high resolution, mainly due to the fact that their computational complexity is quadratic in relation to the image size. This means that processing images using a ViT becomes inefficient for high-resolution tasks. Additionally, the fixed scale tokens in ViTs may perform worse in some applications since they might struggle for tasks involving visual features of different scales. As a successor of the ViT and to address the mentioned problems of it, the Swin Transformer was proposed by Liu et al. (47). It achieves better performance on many computer vision tasks and is designed with greater efficiency. The Swin Transformer ensures a more flexible and sophisticated processing of images by taking into account the variable scales of visual features rather than handling visual data uniformly. These qualities may also be beneficial to our automated pain detection task. Facial expression, including pain-related expressions, may differ significantly in size and intensity. For example, a small facial twitch or a tiny movement of the eyebrow could be signs of pain. Because of ViTs' fixed scale token limitation, these nuances may not be fully captured. Since the Swin Transformer can adapt to different visual scales, it may be able to detect pain more accurately and precisely by registering even the smallest movements and facial expressions. The following chapters will go deeper into the Swin Transformer's technical overview, discussing its design and functions. We will also discuss relevant works and contributions in general and in the field of APA.

4.2.1 Technical overview

An architecture overview of the Swin Transformer is given in Figure 6. The Swin Transformer starts by dividing an input RGB image into non-overlapping patches, just like its predecessor, the ViT. Every one of these patches is seen as an individual token. The features of a token are obtained by concatenating the RGB values of its raw pixels. Selecting a 4x4 patch size results in a feature dimension of 4x4x3 = 48 for each patch. This raw-valued feature is projected by a linear embedding layer to an arbitrary dimension, represented mathematically by 'C'. The first step of the procedure includes this transformation via linear embedding, and two Swin transformer blocks come after it. Looking at the architecture in more detail shows that the "Patch Merging" block and the "Swin Transformer Block" are the two components of each stage that comes after the first stage. These blocks represent the two key concepts, hierarchical feature maps and shifted window attention

which ensure the adaptive and efficient processing of an image.

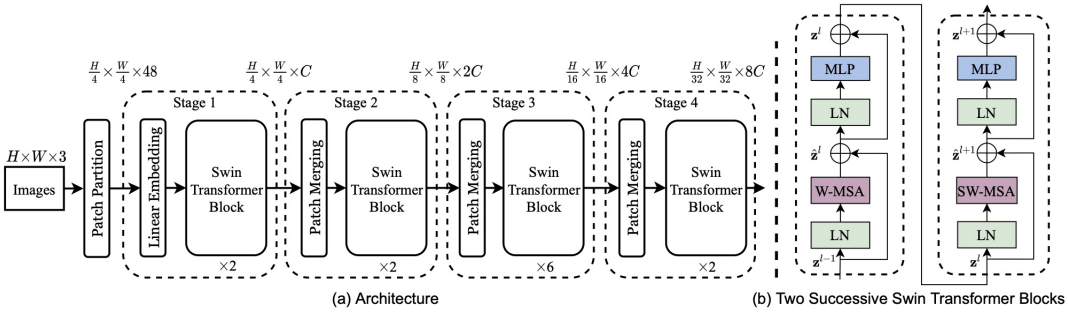


Figure 6. *Technical overview of the Swin Transformer architecture (tiny) (47)*

Patch merging. The patch merging block in the Swin Transformer architecture represents how the hierarchical feature maps concept is implemented in the Swin Transformer. The idea behind the concept is to create a hierarchy of feature representations from the input data by gradually merging and down-sampling successive feature maps. This allows us to capture both, fine-grained details and larger abstractions. While ViT is primarily based on consistent low-resolution feature maps across its architecture, the hierarchical structure of the Swin Transformer allows it to learn a wider range of visual features. Looking more detail into the implementation of this concept, downsampling the feature maps by a factor of 'n', the patch merging process concatenates the features of each group of n x n neighboring patches. Referencing Figure 7, the patch merging mechanism can be divided into three distinct steps for an example where n=2 and 4x4 patches (a): First, the input image patches are separated into multiple groups, with 2x2 neighboring patches in each group (b). Second, the 2x2 patches are concatenated along the depth dimension inside each group (c). After each group's depth-wise stacking is completed, the results are combined to create the downsampled feature map (d). After this operation, the input is downsampled, changing from its original form of H x W x C to (H/n) x (W/n) x (n^{2}*C).

Swin Transformer Blocks. Further to the patch merging, the Swin Transformer presents another block designed to enhance its attention mechanism. This can be achieved through changing the ViT's original attention module. The Swin Transformer block is made up of two separate blocks, as Figure 6 (b) shows. The structure of each of these subunits consists of a normalization layer, an attention module, another normalization layer, and an MLP layer. The difference appears in the kind of attention module that is used. A window multi-head self-attention (W-MSA) module is included in the first transformer block, and a shifted window multi-head self-attention

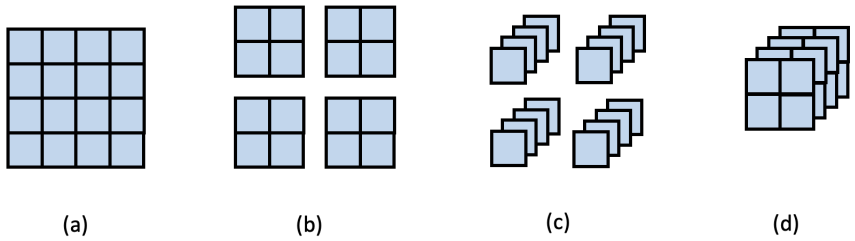


Figure 7. *Patch merging process*

(SW-MSA) module is used in the second block.

While the standard multi-head self-attention, used by ViT, operates under a global self-attention - considering interactions between all patches against one another - the W-MSA version restricts this attention to each respective patch window (see Layer 1 in Figure 8). This specific method ensures that the computational complexity of the W-MSA is linear in respect to the number of patches, or, alternatively, the size of the image. Compared to the quadratic complexity of the conventional multi-head self-attention, this offers a notable optimization. Nevertheless, there is a specific problem with using window-based self-attention alone. There is potential that reducing self-attention within each window will reduce the network's overall modeling performance. To be able to get around this constraint, the Swin Transformer uses the second block — which has a SW-MSA module — after the first W-MSA module.

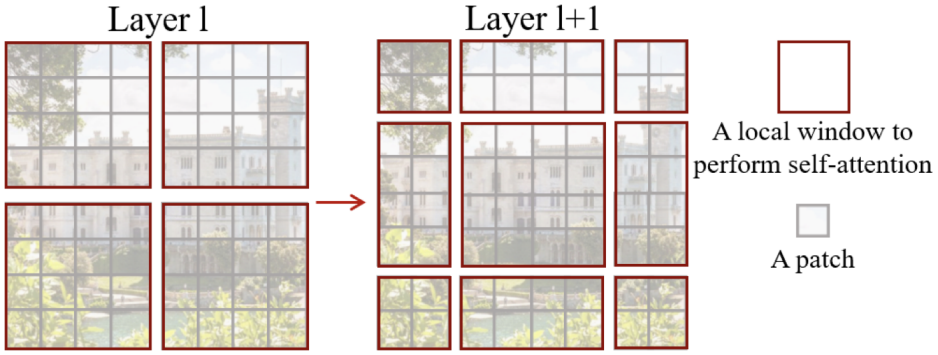


Figure 8. *The shifted window approach for self-attention computation (47)*

The SW-MSA provides cross-window connections as part of its procedure. This is achieved by shifting the windows by a factor of W/2, where W is the window size, towards the lower right corner (see Layer 1+1 in Figure

8). But this moving procedure leaves patches that are without a window. A so-called cyclic shift approach is implemented into the Swin Transformer to compensate for the imbalance produced by the shift. Here, the remaining patches are moved to windows without a full patch set. Notably, with this cyclic shift, a window may now include patches that were not adjacent to the original feature map. In order to preserve the integrity of the attention mechanism, self-attention is limited to neighboring patches by masking these areas. The cyclic shift process is shown in Figure 9.

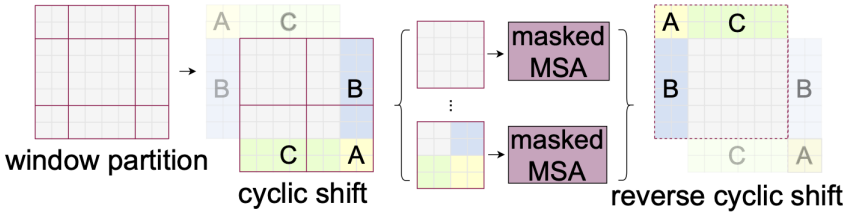


Figure 9. *Cyclic shift process (47)*

4.2.2 Related work using Swin Transformers

The Swin Transformer represents a significant advancement, building upon the foundation set by the ViT. This is not just theoretical; it has been also proven in practice, for example in the original paper (47), the Swin Transformer achieved a comparable state-of-the-art Top-1 accuracy of 87.3% on the ImageNet dataset.

Shifting the focus to the field of facial expression recognition, the Swin Transformer has shown promising potential as well. The study by He et al. (32) is particularly notable, using the Swin Transformer to classify both, macro and micro facial expressions. This research highlights the model's capability to capture fine-grained facial expressions, as shown by its better performance over the MEGC 2022 spotting baseline and comparable results in the MEGC 2021 task. Additionally, the research conducted by Kim et al. (41) explores a multi-modal Swin Transformer approach applied to the Aff-Wild2 dataset, which integrates visual, temporal, and audio data for facial expression recognition. Although their approach does not only include facial expressions, the potential of the Swin Transformer could be shown with a good F1 performance of 0.357 on the Aff-Wild2 dataset.

In the more specific automated pain assessment, the role of the Swin Transformer, though in its first stages, is beginning to be recognized. One of the few studies is from Nerella et al. (59), demonstrating an end-to-end

framework employing the Swin Transformer for identifying pain-related AUs within the Pain-ICU dataset. The Swin Transformer's strength is highlighted by its outperformance of the ViT, achieving an F1-score of 0.88 and an accuracy of 0.85. Further research in this area is made in a study by Yuan et al. (94) focused on pain intensity recognition from partially occluded facial expressions, a common challenge in intensive care settings. The study's approach, involving the pre-training of the Swin Transformer with masked faces in the UNBC dataset, resulted in a good performance in binary and four-level pain intensity measurement tasks with accuracies of 97.38% and 95.25%.

4.3 Video Swin Transformer

The Video Swin Transformer published by Liu et al. (48) is an extension of the original Swin Transformer architecture, specifically adapted to handle video data. Building upon the foundations given in the original Swin Transformer paper (47), this variant introduces additional mechanisms for managing the temporal dimension in video data. Due to the additional temporal dimension, the amount of processing data is significantly higher than in the 2D version, making the locality inductive bias in its self-attention module even more important. Compared to approaches that use a global self-attention module on video data, such as in the ViViT, the VST is very beneficial regarding the computational and memory aspects. In the subsequent subchapters, the technical overview of the architecture is given, pointing out the differences to the original Swin Transformer.

4.3.1 Technical overview

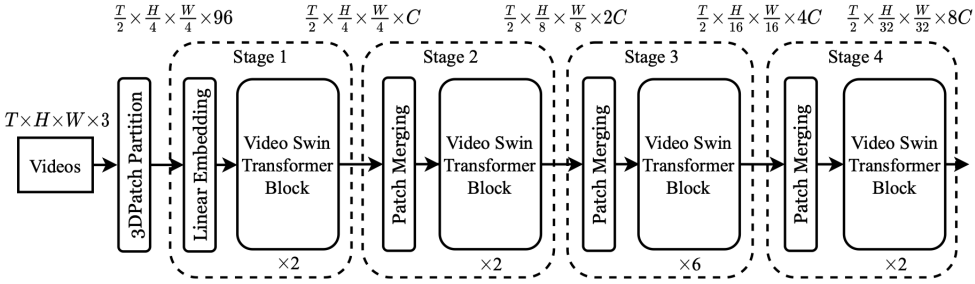


Figure 10. *Technical overview of the Video Swin Transformer architecture (tiny) (48)*

An overview of the Video Swin Transformer architecture is presented in Figure 10. In terms of the input, the Video Swin Transformer processes

input videos defined by dimensions T × H × W × 3, where T represents the number of temporal frames and H × W × 3 specifies the pixel dimensions. Each 3D patch of size 2 × 4 × 4 × 3 is considered as an individual token, leading to a total number of T/2 × H/4 × W/4 3D tokens, with each patch includes a 96-dimensional feature. These features are then as well linearly embedded into a dimension denoted by 'C'. It is important to note that the patch merging does not downsample in the temporal dimension and, hence, focuses only on the spatial level. The patch merging process in each stage otherwise works identically to the original Swin Transformer. Furthermore, the architecture introduces new MSA-3D blocks where the W-MSA and SW-MSA of the original structure are replaced by their 3D counterparts. In this setup, the multi-head self-attention is applied to non-overlapping 3D windows.

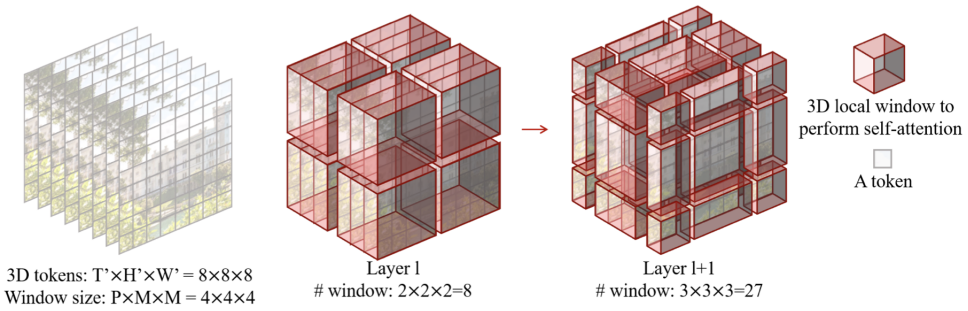


Figure 11. *Example of 3D shifted windows (48)*

As an example, consider an input consisting of 8×8×8 tokens. If we apply a window size of 4×4×4, it results in a total of 2×2×2=8 windows in a given layer l, where each window independently undergoes MSA. This step is visually represented in Figure 11. Regarding the shifted window mechanism, the 3D windows are shifted along the temporal, height, and width axes. The shift is quantified by the formula (P/2, M/2, M/2) tokens, where P and M represent dimensions of the shift in the temporal and spatial axes, respectively. For instance, in layer l + 1 of the example, the windows are shifted by (2, 2, 2) tokens, leading to a new configuration of 3×3×3=27 windows. However, to maintain computational efficiency and neighboring patches, the same masking technique is applied as used in the original Swin Transformer. Following this strategy, the number of windows for computation remains with 8 windows.

4.3.2 Related work using Video Swin Transformers

In terms of related work, the original paper of the VST (48) demonstrates its effectiveness on benchmark datasets and compares it with the ViViT model. For example on the Kinetics 600, the Video Swin Transformer could reach a state-of-the-art accuracy of 86.1%, which is an improvement compared to the ViViT on this dataset (85.8% accuracy). However, there appears to be a lack of research on applying the VST to facial expression recognition or automated pain assessment tasks, highlighting the need for further exploration in this area.

4.4 Cross-dataset validation and generalization

In data-driven research, particularly in the development of machine learning models, a common practice is to use a single dataset where both training and test data originate from the same source domain. This way, however, might create a significant bias since models that are trained and optimized on a particular domain or context might not be able to generalize well across multiple datasets for the same task. Recognizing this challenge, the cross-dataset survey by Zhang et al. (95) attempts to address it by giving an overview of different cross-dataset generalization methods. Cross-dataset validation is a notable technique among these methods. Using this technique, a model trained on one dataset (training dataset) is tested against other datasets (validation datasets) that were not used for the model's training but are from a different domain or context. This procedure is important for assessing the model's robustness and generalizability in different settings, and for making sure the model continues to function well and be reliable even when it encounters data that is not from the training set.

Cross-dataset validation is also important in the specific case of automated pain assessment. Different situations and causes of pain may appear as different types of pain, such as shoulder-related pain (UNBC McMaster) versus heat-related pain (BioVid). Furthermore, datasets can have variations not only in pain contexts but also in their setup and design. While the UNBC McMaster dataset is a clinical dataset, primarily consisting of video recordings of patients with shoulder pain under clinical examination conditions, the BioVid heat pain dataset represents an experimental setup, where participants are exposed to controlled heat stimuli to trigger pain responses. Such variability is a significant challenge for automated pain assessment systems, making cross-dataset validation an important part of assessing their adaptability to different pain contexts. Gkikas and Tsiknakis (31) presents in their survey a significant issue in the field of automated pain detection -

most research is based on a single dataset, leading to a lack of information about the robustness and domain generalization of proposed approaches.

To address these gaps, some studies have investigated cross-dataset validation in the pain assessment field. For instance, Dai et al. (17) experimented with combining pain and emotion-detection datasets to develop a real-time pain assessment system with enhanced generalization capabilities.

Furthermore, the research conducted by Othman et al. (62) highlights the need for more comprehensive studies to enhance the robustness of pain assessment models. The authors advise using multiple datasets in the facial pain detection field due to the reasons mentioned earlier. In their study, they investigate the generalization capability of several pain recognition models by applying cross-dataset validations with two benchmark pain datasets, BioVid and X-ITE. Their results demonstrated that their models performed well in cross-dataset validation. However, it is important to note that both datasets are experimental datasets.

Another relevant research about cross-dataset validation by Prajod et al. (63) focused on the challenges when testing models trained on clinical (UNBC McMaster) on experimental (BioVid) pain datasets and vice versa. The clinical pain model, trained on the UNBC McMaster, although performing robustly within its dataset, demonstrated a considerable decline in performance when evaluated against the experimental BioVid dataset. In contrast, the experimental pain model showed consistent performance across both datasets, suggesting its better generalization capabilities.

4.5 Interpretability and explainability

The interpretability of models and the explainability of their predictions are important research fields in regard to the trust and reliability of applications, especially within the medical sector. Interpretability refers to the extent to which the internal mechanics of a machine or deep learning model can be understood by humans (40). Explainability, on the other hand, involves the ability to explain the outcomes of these models in human-understandable terms (57). In the medical field, where decisions can have significant consequences on patient health and treatment outcomes, the importance of these concepts is significant. Trust in AI systems by healthcare professionals and patients is based on the transparency and comprehensibility of the decision-making process.

4.5.1 Model-agnostic vs. model-specific explainability

Deep learning systems, such as VST, are often not inherently interpretable. This limitation leads to a greater focus on the explainability of specific predictions from these models. When addressing explainability, a distinction is made between model-agnostic and model-specific methods. Model-specific methods are tailored to the architecture of a particular model, providing insights into its unique decision-making process. For instance, in transformer-based models, attention visualization is a common approach. By visualizing the attention maps, which show how different parts of an input image receive varying degrees of attention from the model, it is possible to identify the important location on the image during decision-making. In contrast, model-agnostic methods offer a more flexible solution, which can be applied across various machine learning models regardless of their specific architecture. However, model-agnostic methods like perturbation-based techniques (28)(29) or SHAP (51) often require significant computational resources to generate heatmap visualizations and may not always offer the same level of accuracy as model-specific methods (13). Consequently, our focus will be on specific transformer-based attention visualization methods, given our use of these models.

In the following sections, we look into the existing methods for extracting attention visualizations specifically tailored to ViT and Swin Transformer models. Furthermore, their application in related work on automated pain assessment is described, followed by their findings from a qualitative analysis of the heatmaps.

4.5.2 Extraction of attention visualization from transformer-based models

Visualization techniques for explaining predictions of transformer-based models are relatively underexplored compared to other architectures. The first attempts to visualize attention in ViT models are adapted versions from techniques originally developed for CNNs, such as **Gradient-weighted Class Activation Mapping** (Grad-CAM) (70) and **Layer-wise Relevance Propagation** (LRP)(6). However, both Grad-CAM and LRP are not specific to transformer architectures and do not explicitly consider the transformer's unique attention mechanisms. While they have been successfully applied in automated pain assessment using non-transformer models (14, 86), their adaptation to transformer-specific models in this field is limited.

A significant advancement in the explainability of transformers, partic-

ularly ViT, came with the work of Abnar et al. (1), who introduced a more effective technique known as **Attention Rollout**. This technique works under the assumption that self-attention within transformers is linearly stacked and uses the aggregation of attention scores across different layers. This method has also been applied in automated pain assessment research. Some work related to automated pain assessment (34) (27) extracted the attention out of the models through Attention Rollout and visualized them. Fiorentini et al.(27), for instance, extracted attention maps from their ViViT model trained on automated pain detection; some of these attention maps are shown in Figure 12.

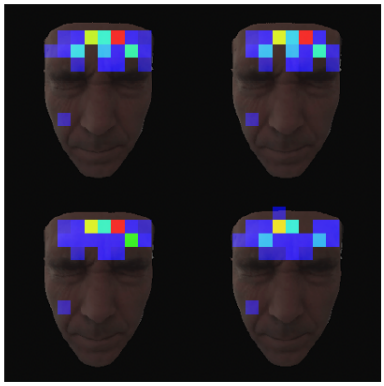


Figure 12. *Example attention visualization of ViViT by Fiorentini et al.(27)*

However, this method, despite its advancements, has limitations. It primarily focuses on the self-attention mechanism of transformers and overlooks other components of the architecture. Moreover, it lacks class discriminative capabilities, meaning it cannot discern whether the contribution of attention is positive or negative. Building upon the foundation laid by this attention visualization technique, Chefer et al. (13) proposed an **improved Attention Rollout**. They include additional transformer-specific elements, using as well gradients of the attention matrix to generate the attention visualization. Their method involves multiplying the relevancy matrix element-wise with the attention matrix gradient and consider the positive value, although in later work by Chefer et al. (12) it was stated that the relevancy matrix can be simply replaced by the attention matrix itself. Furthermore, Chefer et al. (13) conducted a performance analysis, employing perturbation and segmentation tasks to evaluate the efficiency of their approach. Their findings indicate a significant improvement over previous methods.

While these methods were mainly centered around the ViT, there is a noticeable gap in the literature regarding the extraction of attention maps

from successors and variations of the ViT, such as the Swin Transformer. Due to the architectural differences, including the hierarchical structure, extracting attention maps through Attention Rollout is more challenging in this case. Nevertheless, the application and adaptation of this improved **Attention Rollout on the Swin Transformer** were approached in the work by Nguyen et al. (61). They investigated two main differences in the architectures that needed to be considered when applying the method to the Swin Transformer. The first step handles the hierarchical structure of the Swin Transformer, which is achieved by using the row average for the merging patch as the value for the merged patch's corresponding row. This allows one to determine the product of the score matrix of two successive layers with differing feature map sizes. Second, the previous Rollout methods focused only on combining the tokens in a linear way and ignored other elements, such as linear transformation, because they treated each token equally. But in the Swin Transformer architecture, layer normalization is used, which means that each token's value is divided by its standard deviation. Therefore, Nguyen at al. (61) considered the statistical differences of each token in normalization layers by dividing each column of the resulting matrix by the standard deviation of the corresponding token. With their adapted variant, they get reasonable attention visualizations for the Swin Transformer, and, furthermore, they applied the adapted version to the ViT model, which also improved the output by reducing noise. To date, there is no known research that has extracted attention visualization from Swin Transformer for automated pain assessment or similar research fields.

4.5.3 Qualitative analysis of attention visualization in automated pain assessment

As mentioned in the subchapter before, related work has already successfully extracted attention visualization maps from prediction in automated pain detection through facial expressions.

For instance, attention maps from specific frames predicted by Fiorentini et al.'s model (27) effectively focus on critical pain-related facial areas such as the forehead, brows, eyes, and cheeks. This focus aligns with specific AUs (AU4, AU6, and AU43) associated with pain expressions, demonstrating that the model is not only learning to identify random facial features but is actually recognizing pain-relevant regions.

4.6 Imbalanced class distribution in datasets

Imbalanced class distribution in datasets is a common challenge in machine learning that should be considered during training models. This imbalance happens when the number of instances of one class significantly outnumbers that of another, resulting in a skewed distribution that can have a great effect on the learning process and decrease model performance. This is particularly problematic in the field of facial expressions, for instance, in the UNBC McMaster dataset, which includes pain expressions. Difficulties such as the rarity of actual pain expressions within a video sample, combined with the high cost and difficulty of accurate frame-level labeling, contribute to the imbalance in this sector. However, numerous techniques, both at the data and classifier levels, were developed to reduce the negative effects of imbalanced datasets. The next subsection briefly describes two common data-level approaches, undersampling and oversampling, as well as their applications in the field. Furthermore, a more recent classifier-level technique, the Focal loss (46), is introduced, along with its potential in the domain of automated pain assessment.

4.6.1 Undersampling and oversampling

Random undersampling Random undersampling involves reducing the number of samples in the majority class. This approach can speed up training by decreasing the dataset size, which is particularly useful for large datasets. It also has significant drawbacks, including the loss of valuable information and the potential to worsen existing imbalances, making it less suitable for extreme cases of imbalance as it the case, for example, in the UNBC McMaster dataset. Nevertheless, it was also applied in studies (9, 67) using this pain-related dataset, but mainly for pain intensity estimation tasks.

Random oversampling In contrast, random oversampling increases the number of examples in the minority class by randomly duplicating them. This method is straightforward to implement and is particularly useful in smaller datasets. Despite its advantages, it can lead to overfitting and may result in a loss of generalization capability.

Nevertheless, recent studies consistently demonstrate the superiority of oversampling in managing dataset imbalances, for example in the work by Buda et al. (11) with CNNs. Oversampling has been shown to effectively mitigate imbalance without causing overfitting and outperforming undersampling and other methods. Another study on the effectivity of over- and undersampling was done by Mohammed et al. (58), which also indicated the

superiority of the oversampling method.

In the field of automated pain assessment, oversampling has been successfully applied in several studies (8, 52) using the unbalanced UNBC McMaster dataset, as well as in the state-of-the-art study by Fiorentini et al. (27) on pain detection. However, comprehensive investigations addressing the specific imbalances of the UNBC McMaster dataset remain scarce and inconsistent across studies.

4.6.2 Focal loss

Looking at the drawbacks of the previous methods, Focal loss (46) presents an alternative to traditional sampling methods. Focal loss modifies the Cross-entropy loss function to focus more on hard-to-classify examples, which are typically from the minority class. By decreasing the relative loss for samples from the majority class and focusing more on minority examples, Focal loss aims to improve the robustness of the model without requiring changes in the class distribution of the dataset.

Mathematically, the Focal loss can be expressed as:

$$\mathrm{FL}(p_t) = -\alpha_t(1-p_t)^{\gamma}\log(p_t) \tag{3}$$

,where p_t is the model's estimated probability for the class with label $y = 1$, α_t is a weighting factor for the class, and γ is a focusing parameter that adjusts the rate at which easy examples are down-weighted.

The method was initially introduced by Lin et al. (46) to address the significant challenge of foreground-background class imbalance in object detection tasks and surpassed the performance of previous state-of-the-art methods. Since its introduction, Focal loss has also been adapted for use in classification problems beyond its original application in object detection. Particularly, it has been successfully applied in medical image analysis, where imbalanced data is a common issue. For instance, in the study by Duyen et al. (45), Focal loss demonstrated substantial promise in enhancing automatic skin cancer classification systems, which also suffer from heavy class imbalances. Similarly, the technique has been employed in lung nodule classification, another area affected by class imbalance. Tran et al. (79), for example, applied Focal loss combined with data augmentation techniques, achieving an accuracy of 97.2%. This result is comparable to other state-of-the-art methods, which highlights the potential of Focal loss to migate the challenge with imbalanced datasets. Despite these successful applications, the use of Focal loss in automated pain assessment, particularly in imbalanced scenarios such as those presented by the UNBC McMaster dataset,

remains unexplored. This gap in the literature suggests a promising area for future research, as Focal loss could be an effective alternative to current techniques such as oversampling. In the scope of this research, this gap is tried to be filled by applying the Focal loss in automated pain detection.

5 Data

To provide an overview of the data used in our research, the two datasets, the UNBC McMaster and the BioVid Heat Pain dataset, are discussed and analyzed in this chapter.

5.1 UNBC McMaster

The UNBC McMaster Shoulder Pain Expression Archive Database (50) was created by researchers from the University of Northern British Columbia and McMaster University. In our research, this dataset plays an essential role in training and evaluating the performance of the Video Swin Transformer and its comparison models. This database is among the most widely used datasets in pain recognition research due to its rich annotations and focus on shoulder-related pain expressions. An example image sequence of the dataset is provided in Figure 13.

Figure 13. *Data example of the UNBC McMaster dataset(50)*

Looking at the dataset organization, it contains video recordings of 25 patients who experienced shoulder pain and participated in active and passive range-of-motion tests for both affected and unaffected shoulders. The participant demographic is balanced with 12 male and 13 female subjects. Notably, the dataset is limited to visual facial expressions and does not include multimodal data. It consists of 200 video sequences with a total of 48398 FACS-coded frames. These frames are annotated with the PSPI scores and include sequence-level self-report and observer measures. Moreover, 66-point AAM landmarks are provided with each image.

However, it presents a challenge with variable frame sizes; in the dataset 15838 frames at a resolution of 320x240 and 32560 frames at 352x240. An important characteristic of the dataset is its imbalanced distribution towards lower PSPI scores; specifically, 82.71% of the frames have a PSPI score of 0, while 17.02% have a score ≥ 1. This imbalance is depicted in Table 1, which details the frequency of each PSPI score range.

PSPI Score	**Frequency**	**Proportion**
0	40029	82.71%
1-2	5260	
3-4	2214	
5-6	512	
7-8	132	17.02%
9-10	99	
11-12	124	
13-14	23	
15-16	5	

Table 1. *Frequency of PSPI Score Ranges in the UNBC McMaster Dataset*

5.2 BioVid Heat Pain

The BioVid Heat Pain Database (83) was developed by the Neuro-Information Technology group at the University of Magdeburg and the Medical Psychology group at the University of Ulm. For our research, particularly for our fourth sub-research question, the dataset allows us to test the generalizability of our model in a different pain context compared to the UNBC dataset.

In more detail, the dataset includes a multimodal collection with controlled experimental pain settings, capturing data on skin conductance, ECG, EMG, EEG, and a multiple-camera setup that incorporates depth information through a Kinect camera. In the collection process, 90 subjects were exposed to heat pain induced at four intensities, with annotations extending from no pain to four levels of pain stimuli. The temperatures for the pain stimulation were individually adjusted to each subject's pain threshold and tolerance, with each of the four pain levels being stimulated 20 times in a randomized order, holding the maximum temperature for 4 seconds. In Figure 14, an example frame is illustrated.

The BioVid dataset is partitioned into five parts (A-E), with each focusing on different setups. Our research focuses on part A due to its inclusion of unoccluded faces, which is relevant for our facial expression analysis.

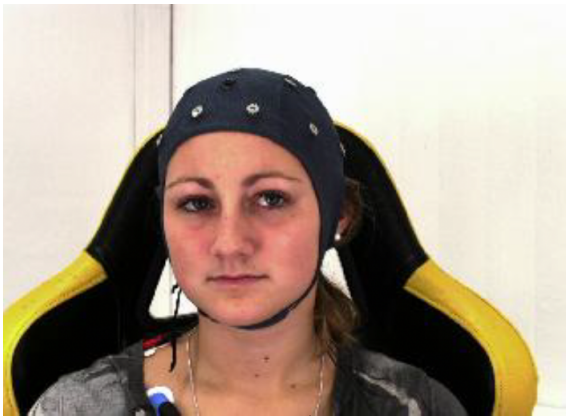


Figure 14. *Data example of the BioVid heat pain dataset(83)*

Moreover, part A consists of pain stimulation without facial EMG, providing frontal video and biomedical signals as raw and preprocessed data. It encompasses 8700 samples from 87 subjects, categorized into 5 classes, with 20 samples per class and subject over time windows of 5.5 seconds.

In a binary pain detection scenario, as is the case in this study, there is a noticeable dataset imbalance towards the pain class, which constitutes 80% of the data, contrasting with the UNBC Dataset.

6 Methodology

This chapter discusses the methodology of the research project, which offers an overview of our automated pain detection pipeline. This includes detailed discussions on preprocessing, the models employed, and the evaluation techniques applied. Following the methodology, we explain the experiments conducted in Chapter 7, illustrating how the methodology pipeline is concretely applied in our research experiments.

To start, Figure 15 visualizes our pipeline. On the left, we have our starting point: a **dataset** containing facial expression samples for pain detection. In our research, we used two datasets, with the primary focus on the UNBC McMaster dataset (50), while the BioVid dataset (83) is partially used for cross-dataset validation. Both datasets undergo a consistent **preprocessing** procedure before being fed into the model. This involves label preprocessing, face frontalization, and subsequence generation. However, the latter is only applied to our spatiotemporal model, the Video Swin Transformer. Chapter 6.1 provides a detailed explanation of these preprocessing steps and highlights their differences between the datasets. After preprocess-

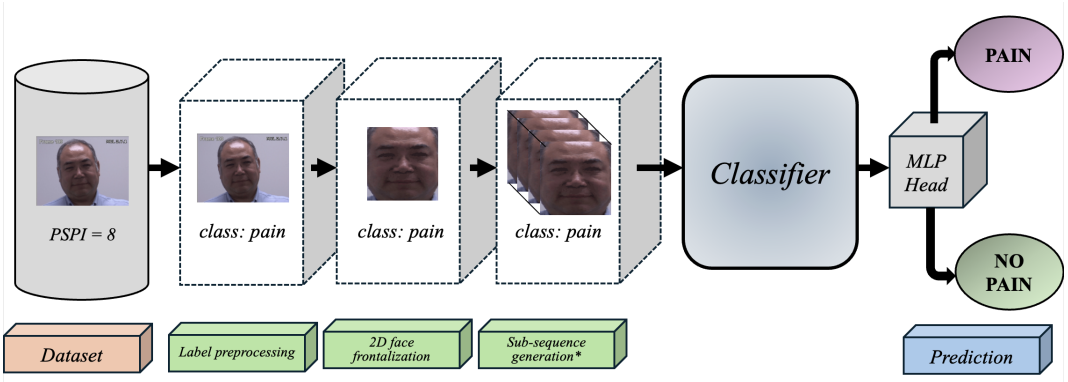


Figure 15. *Pipeline of our automated pain detection system*
**only applied in VST classifier*

ing, the core of our approach lies in the **classifier model**. We examined three architectures: the Video Swin Transformer, the Swin Transformer, and the Vision Transformer. Among these, the Video Swin Transformer is the central model in this research. Chapter 6.2 offers a more detailed technical description at the implementation level and outlines the specific pipelines for each of these three models. The outputs from these models are passed to a MLP classification head, which determines whether the facial expression indicates pain or no pain. This classification head integrates an adaptive average pooling layer, followed by a linear layer responsible for generating the final output. To **evaluate** our approach, we used a mix of quantitative and qualitative methods, described in Chapter 6.3. These methods allowed us to thoroughly assess the effectiveness, generalizability, and explainability of our automated pain detection pipeline.

6.1 Preprocessing

As shown in the preprocessing part of the pipeline in Figure 15, we undertake three steps: 2D face frontalization, label preprocessing, and sub-sequence generation. These preprocessing steps are important to ensure that the data is in a suitable format for training our models. In the following, the preprocessing steps are explained, including the face frontalization process, which is applied for both datasets in the same way, and the remaining preprocessing, which is further described for each dataset separately.

6.1.1 2D face frontalization

To standardize the orientation of faces and make them consistent across samples, face frontalization is applied. In this research, the 2D face frontalization technique provided by RetinaFace (19) with its re-implementation (73)(74) is applied to all our sample frames for both datasets. Unlike preliminary experiments with 3D frontalization using PRNet (26), which showed noise and distortions in cases where the face is in a unfavorable or covered position, RetinaFace's 2D frontalization aligns faces more accurate in these challenging scenarios. Afterwards, the frames are normalized and resized to 224x224 pixels, which ensures uniformity across the samples after the face frontalization. Examples from UNBC McMaster and BioVid databases after the processing is given if Figure 16.

Figure 16. *UNBC (left) and BioVid (right) samples after preprocessing*

6.1.2 Preprocessing UNBC-McMaster

Fold Division. As described in Chapter 6.3.1 in more detail, for evaluating the models, a five-fold cross-validation is performed, with each fold including samples from five participants. To ensure consistency and comparability between the folds, they should have similar class distributions between each other. One division that fulfills this was determined by Fiorentini et al. (27) and the specific class distributions for each fold are presented in Table 2. Due to the similar class distributions and for comparability reasons, this division scheme was preserved.

Data Cleaning. In addition to fold division, data cleaning was conducted to enhance dataset quality. Noisy or unusable frames, such as some that are entirely black, were identified and removed from the dataset.

Label Preprocessing. The original PSPI scores in the UNBC annotations are transformed into binary labels: 0 for no pain (PSPI = 0) and 1

Fold	Label 0	Label 1
1	79.15%	20.85%
2	80.41%	19.59%
3	83.98%	16.02%
4	82.58%	17.42%
5	85.52%	14.48%

Table 2. *Fold division of UNBC McMaster by Fiorentini et al. (27)*

for pain (PSPI > 0).

Subsequence Generation. For the VST models, subsequences of frames are generated to capture temporal dynamics. Unlike concatenating frames in 2x2 grids using the uniform frame sampling technique, as done in previous work by Fiorentini et al. (27), we stack frames along an additional temporal axis following the tubelet embedding. Furthermore, a temporal depth of four frames is chosen, as previous research, such as the ViViT approach by Fiorentini et al. (27), suggests that a sub-sequence length of four frames may be adequate, considering the minimum duration of AUs. Regarding the label, it is determined by the last frame of each subsequence. Furthermore, frames at the video's beginning are artificially extended by duplicating the first frame to maintain temporal depth.

6.1.3 Preprocessing BioVid

Data Cleaning. As well as for the UNBC dataset, first, data cleaning steps were performed. Initially, 20 participants were removed from the dataset as recommended by the publisher of the dataset (83), resulting in a dataset comprising 67 subjects. Additionally, a small amount of noisy data, such as frames where RetinaFace failed to identify a face due to occlusion, were excluded.

Label Preprocessing. Unlike the UNBC dataset, the annotations in the BioVid dataset are not on a frame level but on a sequence level. These annotations include one no pain level (BL1) and four levels of pain (PA1-4). In our preprocessing, we assigned the class 0 (no pain) to the BL1 samples and considered only the two highest pain levels (PA3 and PA4) as the pain class. This decision was based on prior experiments by Yang et al. (93), which indicated less differentiation between the first pain levels and the no pain class. Therefore, the remaining pain levels (PA1 and PA2) were not considered in our experiments.

Subsampling of Videos. Given that the BioVid dataset serves as the test set in our cross-dataset validation and is not used for training, not all

frames of each sequence were required. Investigation by Werner et al. (88) into subsampling each sequence of the BioVid dataset revealed that pain activity typically begins about two seconds after the temperature plateau is reached. Therefore, for videos of pain levels PA3 and PA4, we selected 3 frames per subsequence from seconds 3, 4, and 5. Frames from second two, where pain activity just started, were not considered. To maintain a balanced class distribution, for the videos of the BL1 level, 6 frames were extracted per subsequence from seconds 0.5, 1.5, 2.5, 3.5, 4.5, and 5.5. In total, each subject contributed 240 samples (3 painful frames per PA3 and PA4 class video, 6 non-painful frames per BL1 class video, 20 videos from all classes per subject), resulting in a dataset comprising around 16.000 samples.

Subsequence Generation. For our spatiotemporal models, subsequence generation for the BioVid dataset follows the same approach as for the UNBC samples. Frames are stacked along a temporal axis, combining the main frame with the previous three frames to capture temporal dynamics.

6.2 Models

In this section, we provide a detailed description of our models on both technical and implementation levels. Our model implementations are based on the official PyTorch implementation, offering various architecture variants and modification possibilities for all our used model types. Moreover, all models were trained on a RTX 3060 GPU.

6.2.1 Video Swin Transformer

In the official paper of the Video Swin Transformer (48), four architecture variants are proposed. The official PyTorch implementation provides three of these variants (tiny, small, base). To choose the best variant for our research project, we needed to consider a trade-off between model size and computational complexity. We decided to use the small variant for our main VST and its variants as it offers the same number of layers as the base model but with 0.5× of the base model's computational complexity, due to its smaller number of channels in the hidden layers of the first stage. For our VST models, a pretrained model trained on Kinetics 400 dataset (38) is used, with a total of 49.8 million parameters. Regarding the input, the patch size is set to 2x4x4 pixels, and the window size is 2x7x7 patches. Furthermore, our VSTs have a temporal depth of four frames, except for the one used in the temporal depth experiment using eight frames. The channel number of the hidden layers in the first stage (C) is set to 96. The model consists of four stages with the number of layers [2,2,18,2], respectively. Each layer

block comprises a Video Swin Transformer block and an MLP head. The model-specific pipeline is presented in Figure 10 from Chapter 4.3.1 providing a technical overview of the VST. However, the figure illustrates the tiny variant of the VST instead of the small variant used.

6.2.2 Swin Transformer

For the original Swin Transformer (47), PyTorch also offers multiple architecture variants, and for consistency and better comparability with our VST models, the small variant was chosen. Our implementation of the Swin Transformer leverages a pretrained model trained on the ImageNet-1K dataset (69), resulting in a total of 49.6 million parameters. Unlike the VST, the Swin Transformer operates solely on spatial data, thus eliminating the need for a temporal component. Consequently, the input configuration for the Swin Transformer consists of a patch size of 4x4 pixels and a window size of 7x7. The architecture of the Swin Transformer resembles that of the VST, with each layer comprising a Swin Transformer Block instead of a Video Swin Transformer Block. The number of layers and the channel number of hidden layers in the first stage remain consistent with the VST model as well as the remaining components of the architecture.

6.2.3 Vision Transformer

Within the PyTorch framework, several implementations of ViT are available, including variants such as base, large, and huge, as proposed in the original paper of the Vision Transformer (20). Given the absence of a small variant, the base variant is selected for our study, although differences in model sizes must be considered during comparative analysis. Similar to our approach with the Swin Transformer, we employ a pretrained ViT model trained on the ImageNet-1K dataset (69). The used ViT uses a patch size of 16x16 pixels, and contains 12 layers, each incorporating attention head and fully connected layers. Following the layers, the model employs a MLP head for classification. Furthermore, when looking at the model size, the model has a total of 86.6 million parameters.

6.3 Evaluation

In this subsection, we describe the evaluation methods for our models, including both quantitative and qualitative evaluation. To begin with, the used five-fold cross-validation process is explained. Following that, the quantitative measurements are motivated and described. In addition, we in-

troduce the statistical test employed to compare performance metrics across models. Lastly, the qualitative evaluation methods are presented, which are mainly relevant for the explainability research.

6.3.1 Five-fold cross-validation

For the training and evaluation procedure, we are using a five-fold cross-validation, similar to the approach used in the study by Fiorentini et al. (27). As mentioned before in Chapter 6.1.2, the dataset is divided into five subsets with similar class distribution containing each five subjects. Each model architecture is trained on four of these folds and tested on the remaining one, resulting in five individual models in total per model architecture. This allows us to systematically assess the model architecture's performance across different subsets of the data. We report performance metrics for each fold individually and also calculate the average and standard deviation across all folds to provide a comprehensive assessment of models performance. This cross-validation strategy helps reduce potential biases and ensures that our models performance represent a general picture across the whole dataset. This approach is especially important in studies involving subject data, where the ability to generalize across different individuals is key and not overfit on specific subjects.

6.3.2 Quantitative evaluation

In the quantitative evaluation of model performance for automated pain assessment, selecting an appropriate metric is important but also not always easy. The choice of metric mostly depends on the nature of the dataset, particularly its class distribution. For the primary dataset considered in this research, namely UNBC McMaster, the presence of unbalanced class distribution restricts the choice of suitable metrics. In contrast, the videos in the BioVid dataset are subsampled in such a way that the class distribution is balanced. Another important aspect of the metric selection is to consider its comparability with related work, suggesting using similar metrics as previous research did. Looking at the previous literature study about related automated pain assessment approches, we highlighted that Fiorentini et al. (27) reported the F1-score and the AUC on their state-of-the-art models. Furthemore, they focused on F1-Score during hyperparameter optimization.

F1-Score The F1-score is the first metric we use for our quantitative analysis. This metric represents the harmonic mean of precision and recall, offering

a balanced approach to assessing a model's performance. The formula for calculating the F1-score is:

$$F1 = \frac{2 \times TP}{2 \times TP + FP + FN} \tag{4}$$

,where TP represents true positives, FP false positives, and FN false negatives.

Given the imbalanced nature of the UNBC McMaster dataset, the F1-score provides a more reliable evaluation than the accuracy metric, which can be misleading in skewed datasets. In this case, accuracy may reflect high performance even when the model mainly predicts the majority class. However, the F1-score balances precision and recall, reducing the impact of class imbalances. Moreover, our models are optimized with the F1-Score, which ensures comparability with the work by Fiorentini et al. (27).

Area Under the Curve (AUC) The AUC, representing the Area Under the Receiver Operating Characteristic (ROC) Curve, is another important metric for our evaluation. This metric provides an aggregate measure of a model's performance across all possible classification thresholds. The AUC is especially valuable in scenarios with imbalanced datasets because it reflects the model's ability to distinguish between classes, regardless of threshold settings.

The AUC is selected for two primary reasons. First, AUC tends to be more stable in imbalanced datasets compared to the F1-score, as evidenced by Jeni et al. (35) indicating that it is less susceptible to dataset skew. Second, AUC allows comparability with related work, such as Fiorentini et al. (27), who also reported this metric alongside the F1-score.

Accuracy While not focusing on accuracy for the UNBC McMaster dataset due to its skewed class distribution, we report it when performing cross-dataset validation on the balanced BioVid dataset. Accuracy, defined as the proportion of correct predictions among total predictions, is calculated as:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{5}$$

,where TP represents true positives, TN true negatives, FP false positives, and FN false negatives.

In balanced datasets like BioVid, accuracy can be a useful metric for assessing overall performance.

6.3.3 Statistical significance tests

To answer some of the project's research questions, model comparison needs to be done. To compare the quantitative performance of two models if they are significant different of each other, a suitable statistical test is required. This study used McNemar's test (55) for all model comparisons. The statistical test was chosen because of its basic assumptions — random samples, independence, and mutually exclusive groups — that align with the facts of the used data. It assumes that each model contains at least 25 samples, as is the case in our studies. The null and alternative hypotheses for this test are as follows:

- H0: *The error proportion on the test set for both classifiers is similar.*
- H1: *The error proportion on the test set for both classifiers is different.*

The null hypothesis H0, that the classifier has a similar proportion of errors on the test set, can be accepted or rejected depending on the significance level of the experiment's chosen alpha value and the test's determined p-value. The null hypothesis can be maintained if the p-value is greater than the alpha. In our case the alpha value was chosen to be 0.05. For the statistical test, the predictions on the test set across all folds are included.

6.3.4 Qualitative evaluation

In this section, we present the qualitative evaluation methods used in the explainability part of our research. Specifically, we focus on how to extract attention visualization maps from the predictions made by our models. These visualizations are important for understanding how our Video Swin Transformer, Swin Transformer, and Vision Transformer models focus on different areas of the input during the decision-making process.

Given that the method proposed by Nguyen et al. (61) is currently the only one applicable to Swin Transformers, we followed the methodology outlined by the authors, focusing on the last two layers of the architecture. In the case of our spatiotemporal variant of the Swin Transformer, our VST, we introduced an adaptation to the method. Specifically, we extracted attention visualizations from all four frames of our temporal depth separately and subsequently averaged them. For our ViT model, we implemented the Attention Rollout technique by Chefer et al. (13), as it works similar to the method by Nguyen et al. (61), but for the ViT architecture. Similar to the other two models, the last two layers were considered for the Rollout techniques applied to our ViT.

7 Experiments

Following the presentation of the methodology for this research project, this chapter covers the experiments that were conducted and how the methods were applied to them. Furthermore, two separate subchapters explain the hyperparameter optimization and the fine-tuning process.

7.1 Overview of experiments

7.1.1 Automated pain detection using Video Swin Transformers

Goal: The primary objective of this experiment is to investigate the performance of the Video Swin Transformer (VST) in automated pain detection, addressing the main research question. Additionally, we aim to explore the hyperparameter space of the VST to evaluate the optimal configuration.

As a starting point of the research, we establish a VST and evaluate its performance on the given classification task. This model will serve as a reference for subsequent experiments.

Our approach follows the general pipeline shown in Figure 15, which incorporates the VST model described in Chapter 6.2.1 as the core classifier. For training and testing of our model we are using the UNBC-McMaster dataset and apply the described five-fold cross-validation. The fine-tuning details are described in the Chapter 7.3. Moreover, hyperparameter optimization with a range of hyperparameters is performed to determine the best configuration for pain detection using the VST. For the evaluation of our model we are focusing on the F1-score and the AUC, addressing the challenges posed by imbalanced datasets.

In the subsequent chapters, the resulting VST model of this experiment is denoted as the "main VST" or "VST-0" to make a distinction between this model and its further variations.

7.1.2 Performance comparison of VST and other model architectures

Goal: Building upon the previous experiment, the goal of this experiment is to gain further insights into the effectiveness of the VST, including the spatiotemporal component, compared to other state-of-the-art models on automated pain detection under the same conditions. This experiment tackles also the main research question as well as the first sub-research question about including the temporal dynamics of pain.

To further assess the performance of our VST model, we conduct comparisons with two different model architectures:

The first model we compare against is the original **Swin Transformer**, which operates on frame-level data, unlike our VST, which operates on video-level data. This comparison allows us to investigate how the spatiotemporal component of the VST influences its performance in automated pain detection and whether the temporal dynamics in a painful face are significant for a correct prediction.

The second model for the comparison is the **Vision Transformer**, a predecessor of the Swin Transformer known for its SOTA performance on automated pain detection in previous literature by Fiorentini et al. (27). Instead of re-using the results of the previous literature, we want to ensure a fair comparison with our other models regarding preprocessing, hyperparameter tuning, etc., and, therefore, train our own ViT under same conditions. Initially, we planned to compare with the ViViT to further understand the differences in architecture types on a spatiotemporal level and how the design variances of the VST affect performance in automated pain detection. However, the only available pretrained models were trained using 32 frames as temporal input. Preliminary experiments using only four frames, as in our VST model, with these pretrained models showed very poor performance. Artificial extension of the frames was considered but deemed infeasible due to excessive training time, exceeding the scope of this research project. Nonetheless, using the ViT as a comparison model can still provide valuable insights. Comparing it with our original Swin Transformer, as both work with single frames rather than sequences, offers an insightful comparison.

The pipeline for both comparison models is similar to the VSTs one, except for skipping the subsequence generation, as both models operate on frame-level data, and using the Swin Transformer (Chapter 6.2.2) and the Vision Transformer (Chapter 6.2.3) as classifiers. Furthermore, all models undergo the same hyperparameter optimization and fine-tuning processes. Similar to the evaluation of the VST, performance evaluation of the comparison models will focus on the F1-score and the AUC. When comparing the models, we use the McNemar test to determine if there are significant differences in performance between the models.

For the following experiments, the main VST and these two comparison models are indicated as our main models. Within the comparison models, the Swin Transformer is denoted as “ST-0”, the ViT as “ViT-0”.

7.1.3 VST with extended temporal depth

Goal: The goal of this experimentation is to gain further insights into the temporal axis of VSTs through the integration of additional temporal pain dynamics. This investigation aims to answer the second sub-research question by analyzing the impact of extended temporal depth on VST model performance.

This experiment explores the impact of temporal depth on the VST performance. Our main VST model operates with a temporal depth of four frames as input. For this variation, we increase the temporal depth to eight frames. This change provides insights into the usage of the length of the sequence and its impact on performance. The choice of eight frames doubles the input length compared to the main model, potentially leading to significant differences in performance while remaining computationally feasible.

Evaluation and comparison methods in this experiment remain consistent with the previous experiments, focusing on the F1-score and the AUC. The McNemar test is employed to determine if there are significant differences in performance between the VST with extended temporal depth and the main VST model.

This variant is in the subsequent sections indicated as "VST-1-TD".

7.1.4 Training of VST using Focal loss

Goal: Another experiment aims to investigate an alternative technique, the Focal loss (46), to deal with imbalanced datasets like the UNBC McMaster. The effectivity of this loss function is tested when training the VST in pain detection scenarios, answering the third sub-research question.

In the previous models, Cross-Entropy loss with oversampling of the minority class is used to cope with the imbalanced nature of the UNBC dataset. Nevertheless, in this analysis the Focal loss, detailed explained in Chapter 4.6, without biased sampling is applied and its performance is compared with the one of our main VST. The applied evaluation and comparison methods are the same as in the previous experiments.

This variation will be referred to as "VST-2-FL" in the sections that follow.

7.1.5 Cross-domain generalizability

Goal: Another essential aspect of this research, which is covered by the fourth sub-research question, is to gain insights into the generalization

capabilities of our main VST and its comparison models (Swin Transformer and ViT).

To address this objective effectively, we conducted a cross-dataset validation experiment to measure the performance of our models on unseen data from a different pain domain. Specifically, we assess how well our best models, trained on the UNBC dataset, performed when tested on samples from the BioVid dataset, where participants experienced a different form of pain, specifically heat-induced pain.

In addition to the variance in pain types, several other factors distinguish the two datasets. Notably, there are disparities in demographic characteristics such as age groups; particularly, participants in the UNBC dataset are older compared to those in BioVid. Furthermore, there are differences in the setups, with BioVid adopting an experimental environment while UNBC relies on clinical settings. Additionally, BioVid participants wear EEG caps, a setup absent in the UNBC dataset.

These differences present an opportunity to gain insights into the generalizability of our models across diverse contexts. By accounting for variations not only in pain stimuli but also in demographic profiles and settings, we aim to provide a comprehensive evaluation of the model's ability to generalize. In addition to the F1 score and the AUC, we also consider accuracy for evaluation, as the BioVid test set we are using is equally balanced regarding classes.

All the training and test combinations are summarized in Table 3.

Table 3. *Cross-dataset validation for testing model generalizability*

Model	Fold	Training Dataset	Testing Dataset
VST-0	1 - 5	UNBC	BioVid
Swin-0	1 - 5	UNBC	BioVid
ViT-0	1 - 5	UNBC	BioVid

7.1.6 Explainability

Goal: The final experiment has been conducted to answer the last sub-research question and aims to investigate the explainability of our main models, and how the focus points from the attention visualization differs between the models.

First, we extract attention maps from individual samples predicted by our main VST, Swin Transformer, and ViT model, following the methods proposed in Chapter 6.3.4. The results of these attention visualizations are

represented as heatmaps indicating where the model pays the most attention. To obtain a general overview of the parts responsible for the models' decisions and the relevance of different parts of the face, we average multiple attention visualizations. Doing this, we categorize the averaged heatmaps into four categories of a confusion matrix: false positive, false negative, true positive, and true negative samples. For each category, we extract 100 sample heatmaps and average them. Furthermore, a qualitative analysis is done, where we further observe, analyze, and discuss these focus points, exploring why they may appear for a specific model, and why they differ. The knowledge of pain-specific AUs and the FACS is also used to support our analysis and conclusions.

7.2 Hyperparameter optimization

The performance of our models is significantly affected by the optimal values of a number of model and training hyperparameters. To determine the best values for these hyperparameters, a hyperparameter optimization process was conducted. Given that we train the models on a five-fold cross-validation setup, it was aimed to identify the best hyperparameters for each fold separately, resulting in five optimal combinations per model architecture. To achieve this, we employed a nested cross-validation framework, with an overview provided in Figure 17. It is appropriate for our five-fold cross-validation setup despite being computationally expensive, and ensures that the hyperparameters are selected unbiased without seeing the test set. This bias can be criticized in previous work by Fiorentini et al. (27), as the best hyperparameters were selected using directly the test set and without a separate validation set.

In the outer iteration of our nested cross-validation, we iterated over the main folds, as outlined in Table 2. Each iteration involved holding one fold as the test set (colored orange) while using the remaining four folds for training (colored green/blue). Within the nested cross-validation, an additional inner iteration involved holding out one of these training folds (colored green) as a validation set (colored blue) for hyperparameter tuning. This approach ensured that our test set remained unseen during the hyperparameter optimization process, thereby preventing bias. For each main fold in the outer iteration, we obtained four scores from the inner iterations, which were then averaged. The hyperparameter combination with the best averaged score for each fold was selected for our final optimized model.

For the hyperparameter optimization process, we utilized the Optuna framework (3), employing the Tree-structured Parzen Estimator (TPE) sam-

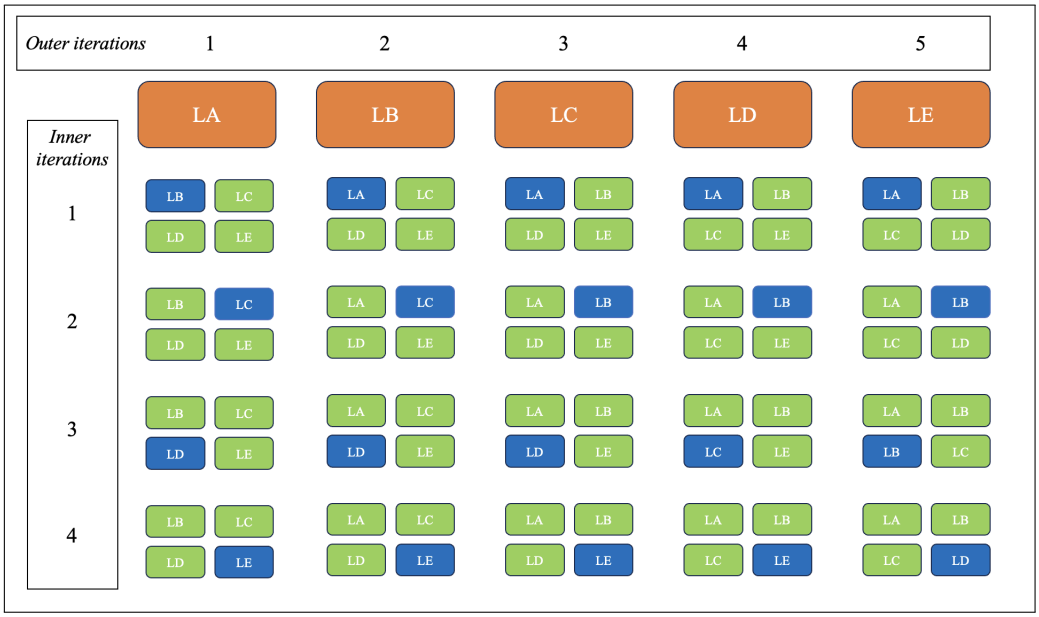


Figure 17. *Overview of hyperparameter optimization through nested cross-validation*

pler. The TPE sampler performs Bayesian optimization on kernel fitting and is recommended for cases with fewer trials and uncorrelated parameters, as is the case here. Due to comparison reasons as this was also done in previous work by Fiorentini et al. (27), the optimization aimed to maximize the F1-score. We conducted the tuning for each main fold with 20 trials, supplemented by preliminary optimization results under the same conditions performed with manual grid search (only for VST-0, ST-0, and ViT-0), resulting in a total of 35 trials for each fold.

Table 4. *Hyperparameter Space*

Hyperparameter	**Valuespace**
Learning Rate	loguniform($1E-05$, $1E-01$)
Weight Decay	0, loguniform($1E-06$, $1E-03$)
Batch Size	choice(16, 32)
Unfrozen Blocks	choice(0, 2, 4, 6, 8)
Alpha	choice(0.6, 0.65, 0.7, 0.75, 0.8, 0.85, 0.9)
Gamma	choice(1.0, 1.5, 2.0, 2.5, 3.0)

Given the computational resources required for the optimization process, we limited the hyperparameter space to specific main hyperparameters and possible values, as detailed in Table 4. These hyperparameters were consistent across all models, with the exception of the Focal loss-specific pa-

rameters, which were included only in the experiment where Focal loss was applied. The key training hyperparameters included learning rate and weight decay for the Adam optimizer, batch size, and the two focal-loss-specific hyperparameters, alpha and gamma. Additionally, model hyperparameters, such as the number of unfrozen layer blocks, were considered. Unfrozen layers indicate the layers that are fine-tuned during training, with 0 unfrozen layers implying that only the classification head is fine-tuned. For example, setting 4 unfrozen layers means that the last four layers of the model are unfrozen and fine-tuned in addition to the classification head.

7.3 Model fine-tuning

Having determined the best hyperparameter combinations through our hyperparameter optimization process, we proceeded with the final fine-tuning of our models on each fold. Experiments conducted beforehand, together with insights from previous work by Fiorentini et al. (27), indicated that our models begin to overfit on the training set after just one epoch of training. Therefore, to prevent overfitting, we decided to train our models uniformly for just one epoch. For the loss function during training, we employed the common Cross-entropy loss, given our binary classification task. Additionally, to address the class imbalance present in our dataset, we applied oversampling of the minority class. However, in the case of the Focal loss experiment, we are using Focal loss instead of Cross-entropy loss, but without oversampling. As briefly mentioned in the previous hyperparameter optimization section, for the optimizer used during training, we employed the Adam optimizer (42) as it is a widely used optimization algorithm. It is often recommended as default optimizer algorithm (68) known for its effectiveness as it requires minimal hyperparameter tuning and offers faster computation time compared to some other optimization algorithms.

8 Results

In this chapter, the results of our experiments are presented, starting with the hyperparameters optimization results for each model. Furthermore, the performance of our main Video Swin Transformer in comparison with the Swin and Vision Transformer is given, followed by an examination of the effects of temporal depth extension and Focal loss in VST. Additionally, the cross-dataset validation results to assess the generalizability of our main models are shown. Finally, we look into the findings from our explainability research, including attention visualization and qualitative analysis.

8.1 Hyperparameter optimization

In Table 5 and 6, we present the optimal training and model hyperparameters for each fold of our models. Additionally, we provide the averaged F1-score on the validation sets.

Model	Fold	Learning Rate	BS	Weight Decay	UB	Val Set F1
VST-0	1	$1E-03$	32	0.0	8	0.51
	2	$1E-03$	32	0.0	2	0.55
	3	$1E-03$	32	0.0	8	0.50
	4	$1E-03$	32	0.0	8	0.51
	5	$1E-03$	16	$2E-04$	6	0.45
ST-0	1	$7E-04$	32	$3E-04$	8	0.47
	2	$2E-04$	16	$5E-04$	8	0.49
	3	$1E-05$	16	$2E-05$	8	0.40
	4	$1E-03$	16	$8E-05$	6	0.46
	5	$3E-04$	16	$1E-04$	4	0.38
ViT-0	1	$2E-04$	32	$1E-05$	8	0.52
	2	$1E-04$	32	$3E-06$	8	0.55
	3	$5E-05$	16	$7E-05$	8	0.56
	4	$7E-05$	16	$2E-06$	8	0.57
	5	$2E-04$	16	$4E-06$	8	0.53
VST-1-TD	1	$1E-04$	16	$2E-06$	0	0.45
	2	$2E-03$	16	$2E-04$	4	0.54
	3	$3E-03$	32	$2E-05$	2	0.45
	4	$5E-04$	16	$1E-05$	8	0.51
	5	$4E-05$	32	$4E-04$	4	0.41

Table 5. *VST-0, ST-0, ViT-0, VST-1-TD models best hyperparameters for each fold (UB = Unfrozen Blocks; BS = Batch Size)*

Briefly looking the resulting hyperparameters, there are almost no significant patterns observable across the models. However, it is noticeable that VST-0, ST-0, and ViT-0 have a tendency for a higher number of unfrozen blocks. Furthermore, in VST-0, 4 out of 5 folds performed the best with the default weight decay of 0. Additionally, the learning rate tends to be lower for ViT-0 and the VST-2-FL compared to the other models.

When looking at the validation set scores, the ViT-0 seems to perform the best out of all the models, followed closely by our main VST-0 model. In contrast, compared to other models in our investigation, ST-0 and VST-2-FL with Focal loss perform relatively worse on the validation sets.

Model	Fold	Learning Rate	BS	Weight Decay	UB	γ	α	Val Set F1
VST-2-FL	1	$2E-03$	16	$1E-06$	0	2.0	0.85	0.34
	2	$2E-05$	32	$2E-05$	4	1.5	0.7	0.50
	3	$1E-05$	32	$3E-04$	4	2.5	0.75	0.46
	4	$3E-05$	16	$8E-06$	6	1.0	0.6	0.44
	5	$1E-05$	32	$6E-06$	2	2.5	0.75	0.43

Table 6. *VST-2-FL models best hyperparameters for each fold (UB = Unfrozen Blocks; BS = Batch Size)*

8.2 Model results

In the following, the results of our main models, including Video Swin Transformer (VST-0), Swin Transformer (ST-0), and Vision Transformer (ViT-0), are presented. After looking at the performance of our model separately, we analyse their outcome in a comparative way between the models.

8.2.1 Video Swin Transformer (VST-0)

Looking at the results of our VST-0 in Table 7, the model achieved an average F1-score of 0.56 with a standard deviation of 0.06, and an AUC of 0.85 with a standard deviation of 0.04. Notably, the first fold reached the highest F1-score (0.65), while the fifth fold exhibited the highest AUC score (0.89). Conversely, the second (0.50) and fourth (0.49) folds demonstrated the lowest F1-scores within the VST-0 model's performance.

Fold	Test Set F1	Test Set AUC
1	0.65	0.87
2	0.50	0.79
3	0.55	0.87
4	0.49	0.81
5	0.61	0.89
Mean	**0.56**	**0.85**
Std	0.06	0.04

Table 7. *VST (VST-0) model results on test set for each fold*

8.2.2 Swin Transformer (ST-0)

The results of the ST-0 model across each fold are shown in Table 8. The average F1-score is 0.53 with a standard deviation of 0.04, while the

average AUC is 0.80 with a standard deviation of 0.02. The best-performing fold for F1-score is the 5th, with a value of 0.60. For AUC, the best results are observed in the 4th fold, with 0.83, followed closely by the 5th fold with an AUC of 0.82. Conversely, the poorest performance across both metrics comes from fold 3, with an F1-score of 0.49 and an AUC of 0.77.

Fold	Test Set F1	Test Set AUC
1	0.52	0.78
2	0.49	0.81
3	0.49	0.77
4	0.53	0.83
5	0.60	0.82
Mean	**0.53**	**0.80**
Std	0.04	0.02

Table 8. *Swin Transformer (ST-0) model results on test set for each fold*

8.2.3 Vision Transformer (ViT-0)

The performance results for the Vision Transformer (ViT-0) model are summarized in Table 9. For this model, the mean of the F1-score is 0.55±0.09, while the mean of the AUC is 0.87±0.04. The best-performing fold in terms of F1-score is the 5th fold, achieving a score of 0.67. This fold also demonstrates the highest AUC of 0.93. The lowest F1-score comes from the 2nd fold, with a score of 0.39, while the corresponding AUC is 0.81.

Fold	Test Set F1	Test Set AUC
1	0.60	0.88
2	0.39	0.81
3	0.56	0.89
4	0.54	0.85
5	0.67	0.93
Mean	**0.55**	**0.87**
Std	0.09	0.04

Table 9. *Vision Transformer (ViT-0) model results on test set for each fold*

8.2.4 Comparison between the models

The comparison among the three models, VST-0, ST-0, and ViT-0, is depicted in Figure 18, that shows two bar plots illustrating the performance

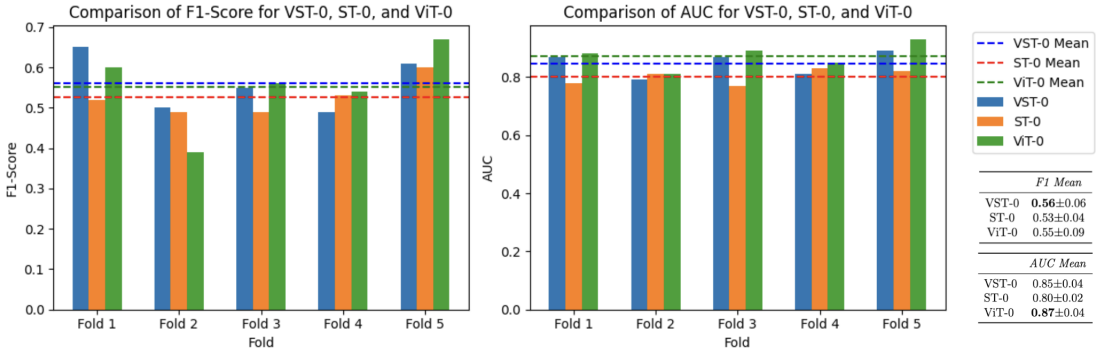


Figure 18. *Performance comparison of VST-0, ST-0, and ViT-0 on each fold*

of each model per fold for both, F1-Score and AUC. Next to the bar plots, the mean value for each model from the previous sections are summarized again in a table.

Looking from a general point of view, our ST-0 model consistently underperforms relative to the other two models, with a mean F1-score of 0.53 and an average AUC of 0.80. This is also evident in three out of five folds where it ranks as the lowest. However, it exhibits the smallest standard deviation under the three models, which usually indicates greater stability. Regarding the other two models, our VST-0 (0.56) and ViT-0 (0.55) are performing on the F1-score relatively close, although our VST model indicates a slightly better mean, suggesting it might be the best model of the three regarding the F1-score. On the other hand, our ViT-0 achieves a slightly higher mean AUC (0.87) compared to VST-0 (0.85), and, therefore, the best-performing of our models when looking at the AUC.

VST-0 vs. ST-0 In a more particular comparison, the VST-0 generally outperforms ST-0 in most folds on both performance metrics. The difference between the two models performance on all folds together is statistically significant, with a p-value of $5.98E-07$ derived from the McNemar significance test. The performance differences may come from the fact that VST-0 incorporates a spatiotemporal component, whereas ST-0 operates only on spatial data. Furthermore, the assumption that including the temporal pain dynamics could be beneficial for detecting pain is further reinforced by the fact that both models were designed and trained under similar conditions.

VST-0 vs. ViT-0 Comparing VST-0 and ViT-0, both models operate on a similar level of performance, although there is a significant difference proven by our statistical test with a p-value of 0.024. As mentioned before, the VST-

0 performs better on the F1-score, and the ViT-0 on the AUC. For imbalanced dataset, the AUC would be probably the more reliable metric (36), although, here, they also indicate comparable performance on that. In particular on fold-level, ViT-0 shows better results across most folds, but VST-0 stands out on the folds one and two, where it clearly outperforms ViT-0 in terms of F1-score. Moreover, the standard deviation for F1-score is lower in VST-0 (0.06 compared to ViT-0's 0.09), indicating that it might be more stable. Looking at these findings, it seems that both models show comparable performance. An important aspect that needs to be mentioned for this comparison is that the ViT-0 is as a base variant implemented compared to our VST-0 (small variant). As a result, the number of parameters is affected and may lead to an unfair comparison. To confirm this, further investigation is necessary, for example, training the VST-0 as well on a comparable base variant, which was not possible in the scope of this research project.

ViT-0 vs. ST-0 In the final comparison between ViT-0 and ST-0, the ViT-0 model distinctly outperforms ST-0 on both F1-score and AUC, with statistical significance of their predictions shown by a p-value of $3.53E-12$. This trend is also given across most folds, that may suggest a potential superiority of the Vision Transformer architecture for automated pain detection. Nevertheless, as mentioned before, this conclusion should be considered with caution due to the differences in model sizes and parameters. Further research and fair comparisons might be needed to draw definitive conclusions about the relative performance of these architectures.

8.3 Temporal depth extension results

Fold	Test Set F1	Test Set AUC
1	0.62	0.88
2	0.56	0.85
3	0.54	0.84
4	0.60	0.87
5	0.62	0.92
Mean	**0.59**	**0.87**
Std	0.04	0.03

Table 10. *Video Swin Transformer with temporal depth extension (VST-1-TD) models results on test set for each fold*

In Table 10, we present the results for the Video Swin Transformer with a temporal depth extension of eight frames (VST-1-TD). This variant of the

main VST model demonstrates a mean F1-Score of 0.59 with a standard deviation of 0.04 and a mean AUC of 0.87 with a standard deviation of 0.03, which indicates a improvement compared to the VST-0 model. The best performance is observed in fold four and five, where the F1-score reaches 0.62, while fold 5 achieves the highest AUC of 0.92. The lowest performance occurs in fold 3, with an F1-score of 0.54 and an AUC of 0.84.

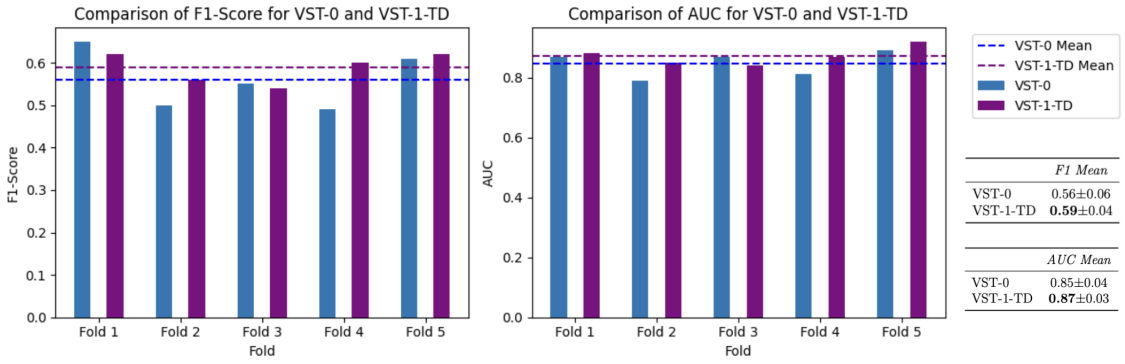


Figure 19. *Performance comparison of VST-0 and VST-1-TD on each fold*

VST-0 vs. VST-1-TD Figure 19 presents similar bar plots as before comparing the performance of VST-0 and VST-1-TD on each fold. The extension from four to eight frames leads to a significant improvement in performance across the majority of the folds, with VST-1-TD generally outperforming VST-0 on both F1-score and AUC. Across the performance mean values, VST-1-TD shows a clear advantage over VST-0. The statistical significance of this improvement is confirmed by the McNemar test, having a p-value of $2.24E-09$. The observed improvement suggests that capturing more temporal depth may allow the model to recognize facial pain dynamics more accurately, resulting in better performance on F1-Score and AUC. Additionally, the VST-1-TD shows a slightly lower standard deviation, indicating greater stability in performance across the folds. All these aligns with the earlier findings from the VST-0 and ST-0 comparison, which indicates that the temporal component is essential for automated pain detection. However, the computing time for the VST-1-TD model is almost double that of the VST-0 due to the extended temporal depth.

8.4 Focal loss VST results

The results for the VST trained with Focal loss (VST-2-FL) are displayed in Table 11. This variant of the VST model shows an average F1-score

of 0.44 with a standard deviation of 0.11, and a mean AUC of 0.77 with a standard deviation of 0.09. More specifically, the best-performing fold for VST-2-FL is fold 5, with an F1-score of 0.65 and an AUC of 0.87. In contrast, the worst-performing fold is fold 2, with an F1-score of 0.34 and an AUC of 0.71. Notably, fold 4 also has a low AUC of 0.71.

Fold	Test Set F1	Test Set AUC
1	0.36	0.77
2	0.34	0.71
3	0.44	0.79
4	0.43	0.71
5	0.65	0.87
Mean	**0.44**	**0.77**
Std	0.11	0.09

Table 11. *Video Swin Transformer trained with Focal loss (VST-2-FL) models results on test set for each fold*

VST-0 vs. VST-2-FL Figure 20 illustrates the performance comparison between VST-0 (the main VST trained with Cross-Entropy loss and oversampling) and VST-2-FL. The VST-2-FL model consistently underperforms compared to VST-0 in almost all folds, and the performance gap is significant. Statistical significance between the two models' performances is confirmed by the McNemar test, with a p-value of 0.0. Besides, the higher standard deviation in VST-2-FL (0.11 for F1-score and 0.09 for AUC) reflects greater variability and lower stability across the folds. The Cross-Entropy loss in combination with oversampling appears to provide a better and more consistent performance.

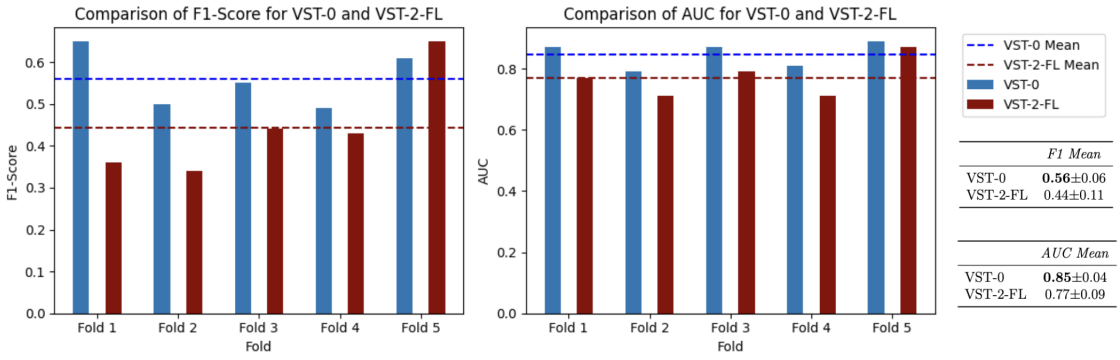


Figure 20. *Performance comparison of VST-0 and VST-2-FL on each fold*

8.5 Comparison with previous work

To evaluate the performance of our models in the context of previous research, a comparison table, Table 12, is given. The table presents the average F1-score and AUC for each of our models and the best-performing models from the work by Fiorentini et al. (27). This previous work is known for its SOTA performance in automated pain detection, using similar fold divisions and experimental conditions. They proposed a Vision Transformer (ViT-1-D) and a Video Vision Transformer (ViViT-1-D), making these models relevant for our comparison. As mentioned in the previous chapters in the comparison between ViT-0 and VST-0/ST-0, these Vision Transformer models differ from our VST and Swin Transformer variants in terms of model size (base variants). Additionally, the ViViT approach uses a grid system rather than a separate temporal axis, providing another point of differentiation in terms of video models.

Model name	F1-Score	AUC
ViT-1-D (27)	0.55 ±0.15	**0.88**
ViViT-1-D (27)	0.55 ±0.13	0.86
VST-0	0.56 ±0.06	0.85 ±0.04
ST-0	0.53 ±**0.04**	0.80 ±**0.02**
ViT-0	0.55 ±0.09	0.87 ±0.04
VST-1-TD	**0.59 ±0.04**	0.87 ±0.03
VST-2-FL	0.44 ±0.11	0.77 ±0.09

Table 12. *Performance comparison of our models with previous work*

Looking at the F1-score metric in the comparison table, the VST-1-TD achieves a new SOTA performance with a mean F1-Score of 0.59, surpassing the ViT-1-D and ViViT-1-D (both 0.55). Our VST-0 also slightly outperforms the previous SOTA with a score of 0.56. Although the previous work reported a SOTA AUC of 0.88, our VST-1-TD and ViT-0 models achieved comparable results with AUCs of 0.87. These scores slightly outperform the ViViT-1-D model (0.86). Across our models, there's a noticeable reduction in standard deviation compared to previous work. The lowest values are found in our ST-0 and VST-1-TD, indicating improved consistency across folds. Given these comparisons, our models show promising improvements over the previous SOTA in terms of F1-score and comparable results in AUC. Additionally, the reduced standard deviation suggests greater reliability. Considering the difference in model sizes, there might be additional potential for the VST architecture to outperform ViT-1-D on AUC if trained on similar

model variants. This suggests that future work could focus on optimizing the VST architecture or re-evaluating the models with a more comparable structure to provide a fairer comparison.

8.6 Cross-dataset validation results

To assess the generalizability of our three main models (VST-0, ST-0, and ViT-0), we conducted a cross-dataset validation. The results from the best models on each fold on the BioVid test set are summarized in Table 13.

	F1 Score			**AUC**			**Accuracy**		
Fold	*VST-0*	*ST-0*	*ViT-0*	*VST-0*	*ST-0*	*ViT-0*	*VST-0*	*ST-0*	*ViT-0*
1	0.54	0.43	0.59	0.60	0.57	0.60	0.58	0.53	0.57
2	0.47	0.64	0.41	0.60	0.59	0.60	0.57	0.53	0.56
3	0.61	0.56	0.41	0.58	0.54	0.59	0.55	0.54	0.57
4	0.42	0.41	0.47	0.60	0.56	0.60	0.55	0.52	0.58
5	0.52	0.38	0.52	0.59	0.58	0.59	0.57	0.53	0.57
Mean	**0.51**	0.48	0.48	0.59	0.57	**0.60**	0.56	0.53	**0.57**

Table 13. *Cross-database comparison on different metrics*

Given that the BioVid dataset has no class imbalance, we also included accuracy along with F1-score and AUC. Regarding the F1-score, the VST-0 model (0.51) is the best on average, outperforming the ST-0 and ViT-0 models (both 0.48). For the accuracy and AUC, the ViT-0 model outperforms the others with an AUC of 0.60 and accuracy of 0.57, although the VST-0 is close behind. The ST-0 model tends to perform the worst across all three metrics. These findings suggest that the VST-0 and ViT-0 models have comparable generalizability, while the ST-0 model underperforms in terms of cross-database validation. This supports the observation that a spatiotemporal approach, such as that in VST-0, has an advantage over purely spatial models like ST-0. Nevertheless, none of the models showed excellent performance in the cross-dataset validation, especially when looking at accuracy.

8.7 Explainability results

In this section, we present the results of our explainability analysis, which uses attention maps to identify the key focus points of each model during prediction. For each of our three models (VST-0, ST-0, and ViT-0),

the outputs are displayed in the form of confusion matrices, which shows the average attention focus for different prediction outcomes (true positive, true negative, false positive, and false negative).

8.7.1 Attention visualization Video Swin Transformer

The confusion matrix in Figure 21 presents the averaged attention visualizations for the VST-0 model. The subject in the background serves only as an example and for facial orientation, rather than representing a specific frame extraction as the attention visualization shows the output averaged from 100 randomly selected samples.

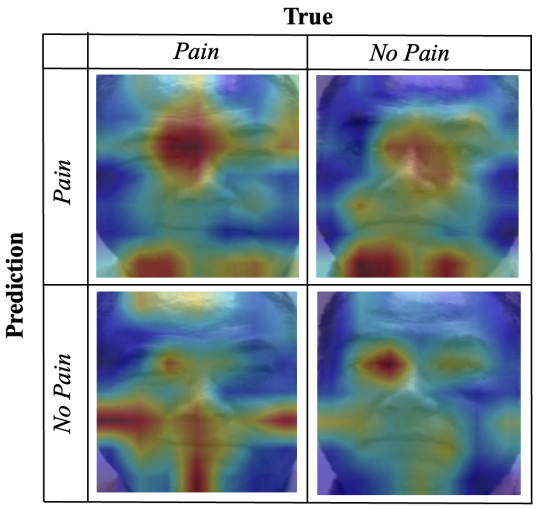


Figure 21. *Attention visualization confusion matrix for VST outputs*

True Positive In cases where the VST-0 model correctly identifies pain, the attention is focused in the middle face region, but particularly around the eyes and brows. The affected pain-specific AUs include: AU4 (lowering of brows), AU9 (nose wrinkling), AU7 (lids tight), AU43 (eyes closed), and slightly AU10 (raising the upper lip). Additionally, there is high focus on the chin region, specifically regions affected during AU16 (lower lip depress), which is not typically pain-specific.

False Positive When the model wrongly predicts pain, the focus remains similar to true positive cases, but with greater attention on the chin region and additional focus beyond the lip corner.

True Negative In correct no-pain predictions, the attention shifts to other areas, focusing more on the eyes (AU43 - eyes closed) and beyond the lip corners. The model also pays attention to the cheek region, which may be related to absence of AU6 (cheek raiser), and around chin.

False Negative When the model fails to detect pain, the attention pattern is similar to true negatives, but with more focus on the chin, cheeks, and mouth regions. There's less attention on AU43 (eyes closed), and more on the forehead.

8.7.2 Attention visualization Swin Transformer

The following Figure 22 presents the attention visualizations for the Swin Transformer (ST-0) model.

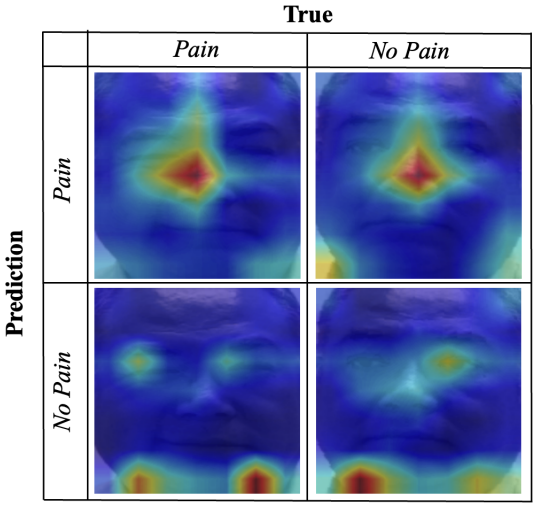


Figure 22. *Attention visualization confusion matrix for Swin Transformer outputs*

True Positive In cases where the model correctly identifies pain, the focus is centered around the nose region, highlighting AU9 (nose wrinkling). Additionally, there is attention towards the brows (AU4 - lowering of brows) and below the eyes (AU7 - lids tight), indicating a strong attention on pain-specific AUs.

False Positive When the model incorrectly predicts pain, the attention visualizations are quite similar to those in true positive cases, with notable focus on regions related to AU9 and AU4. Interestingly, there is also a

slight focus on the left and right corners of the image, which might indicate background noise, potentially leading to incorrect predictions.

True Negative In correct no-pain predictions, the model's attention shifts to the eyes region (AU43 - eyes closed) and slightly to AU9 (nose wrinkling). The primary focus, however, is on the side chin region, indicating regions with an appearance change during AU20 and AU16, which are generally not associated with pain.

False Negative In cases where the model fails to detect pain, the focus is similar to true negatives, but with less attention on the eyes and no focus on the nose region.

8.7.3 Attention visualization Vision Transformer

The explainability results for the Vision Transformer are given in Figure 23, showing the corresponding attention visualization confusion matrix.

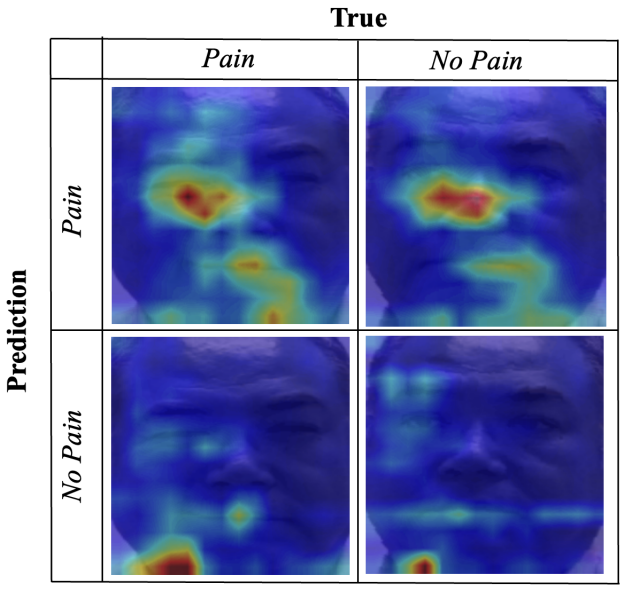


Figure 23. *Attention visualization confusion matrix for ViT outputs*

True Positive In cases where the ViT-0 model correctly identifies pain, the attention is primarily focused on the nose region, corresponding to the pain-specific AU9 (nose wrinkling). The focus also spreads towards the eyes,

particularly AU43 (eyes closed) and AU7 (lids tight). Notable attention is observed in the mouth, lip corners, and chin regions, covering AU14 (dimpler), AU15 (lip corner depressor), and AU16 (lower lip depress). Slight attention is also observed in the brow region, indicating AU4 (lowering of brows), but this is less significant.

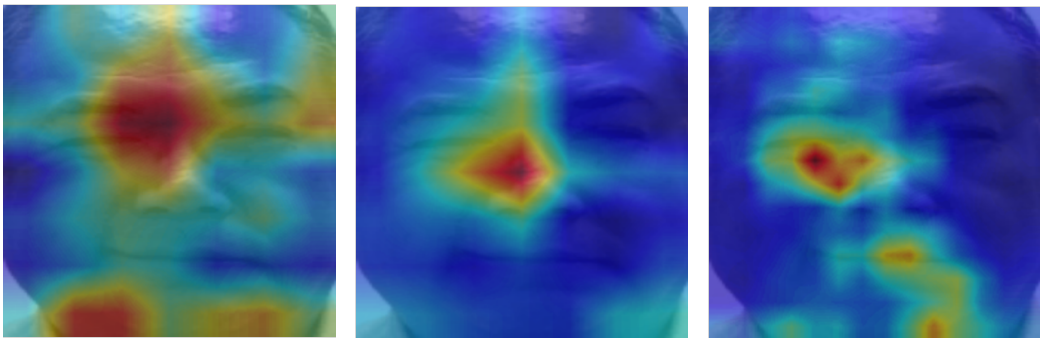
False Positive In cases where the model incorrectly predicts pain, the attention visualizations are similar to the true positive cases, but with slightly more attention on the eyes and nose regions and less on the mouth and chin regions. The brow region is not significantly focused on in these cases.

True Negative In correct no-pain predictions, the focus is mainly on the very low side of the chin region, which is relevant to AU16 (lower lip depress), although this could be background noise in some samples. There is also slight attention in the brow area, indicating AU4 (lowering of brows), as well as around the eyes with AU7 (lids tight), cheeks with AU6 (cheeks raising), and the mouth and lip corners.

False Negative When the model fails to detect pain, the attention is similar to the true negative cases, but with a much stronger focus on the chin region, related to AU16 (lower lip depress), but also background information. The brow region is absent in these cases.

8.7.4 Attention comparison of true positives between the models

This chapter presents a comparative analysis of our three model attention maps, focusing on true positive cases. A composition of these maps from all three models can be seen in Figure 24.



(a) VST *(b) Swin Transformer* *(c) ViT*

Figure 24. *True positive attention maps*

All three models demonstrate that they actually learned to use regions and AUs associated with pain to predict the pain class. The VST model extends its focus across the middle face region, particularly concentrating on the nose and eyes. It also spreads its attention towards the mouth and forehead, and notably, includes the chin region. The Swin Transformer shows a similar attention distribution, centered around the nose, brows, and eyes. However, it does not significantly extend towards the mouth region, which shows a more localized focus compared to VST. In contrast, the ViT shares similar attention patterns with the VST, especially around the nose and eyes and extends similarly towards the mouth and chin regions.

Regarding AUs, the attention mechanisms of all these models share a focus likely corresponding to the following pain-related AUs:

- AU4 - lowering of brows
- AU7 - lids tight
- AU9 - nose wrinkling
- AU43 - eyes closed (mainly for VST and ViT)

Furthermore, the VST and the ViT also together pay attention to the mouth region, indicating potentially the AU10, which is a raise in the upper lip. Interestingly, it seems that these two models also pay attention towards non-pain-specific regions like the chin region (AU16 - lower lip depress).

9 Discussion and Limitations

This chapter focuses on the discussion and limitations of our research. Firstly, the key findings, interpretations, and implications for each research question are discussed, followed by the research limitations and ideas for future work.

9.1 Research questions

How do Video Swin Transformers perform in the automated assessment of pain through facial expressions? In this study, we investigated the efficiency of Video Swin Transformers for the automated detection of pain using facial expressions. This was motivated by the model's potential to capture different scales and nuances of pain dynamics while remaining computationally efficient at high resolution when compared to Vision Transformers. Our findings indicate that VST models, particularly when including

an extended temporal depth, can indeed deliver state-of-the-art performance in this domain. The results demonstrated that the main VST model, designated as VST-0, achieved an F1-score of 0.56 and an AUC of 0.85. Further enhancements in model performance were observed with our optimized VST model (VST-1-TD), which used a temporal depth of eight frames, achieving a higher F1-score of 0.59 and an AUC of 0.87. When compared to the reproduced Vision Transformer model (ViT-0) from prior state-of-the-art research, both our VST-0 and VST-1-TD models showed improvements in F1-score, with the VST-1-TD also surpassing the ViT model in terms of AUC. Moreover, our results also highlight the significant outperformance of the Swin Transformer model ST-0 by our VST models, demonstrating the superiority of VST over its spatial counterparts in handling the dynamic aspects of facial expressions associated with pain.

Building on these findings, it is evident that the best performing VST models not only met but exceeded the benchmarks set by all comparison models in terms of both F1-score and AUC. This reinforces the idea that the architectural characteristics inherent to Video Swin Transformers - specifically, their scaling principle and the inclusion of temporal information - significantly impact the detection of nuanced and dynamic pain expressions.

Furthermore, our analysis indicates that the VST models, VST-0 and VST-1-TD, were capable of outperforming the previous state-of-the-art model reported by Fiorentini et al. (27), which had an F1-score of 0.55 and an AUC of 0.88. Our models not only surpassed this in F1-score but also delivered comparable results in terms of AUC. This comparison as well as with our reproduced ViT-0 model, however, should be contextualized within the framework of model sizes and capacities. The Vision Transformer models used in previous studies were typically based on a larger 'base' model, in contrast to our 'small' model variants for both the VST and Swin Transformer, introducing a potential difference in model capabilities due to the deviation in the number of parameters. This difference suggests that the superiority of our VST models might be even more significant if a fair comparison were made with equivalent model sizes.

A critical examination of the performance differences between the Vision Transformer (ViT-0) and Swin Transformer (ST-0), both operating at the spatial level, suggests that at first glance, the Swin family architecture does not necessarily have clear superiority over the ViT. Nevertheless, this comparison must be seen differentiated by considering as well the significant differences in model sizes, which requires further research to have a fair and balanced evaluation.

However, the improved performance of the Video Swin Transformer

models, especially looking at the VST-1-TD, holds considerable implications for the field of automated pain detection systems. This study represents the first known application of VST technology on this task, achieving state-of-the-art results in terms of F1-score and offering performance that is comparable with state-of-the-art in terms of AUC. The successful implementation and good results of the VST models highlight the architecture's potential and establish it as a promising approach for this specific task. One of the most significant implications of our findings is the validation of the VST architecture for effectively capturing both temporal dynamics and fine-grained spatial details in facial expressions associated with pain. Furthermore, the given computational efficiency of VSTs for high resolution data suggests that they are not only better at dealing with the nuances of pain detection, but also are more suitable for integration into real-time systems. The implications of the findings are not only limited to the field of automated pain detection, but also can be used in related fields, like pain intensity estimation or any automation working with facial expressions.

How does incorporating temporal dynamics of pain at the video-level impact the performance of automated pain detection compared to solely frame-level analysis? To address the first sub-research question regarding the impact of incorporating temporal dynamics at the video-level, we focused on a comparative analysis between the Video Swin Transformer (VST-0) and the Swin Transformer (ST-0). These models share a similar architectural foundation but differ in their handling of temporal components. Our findings show a notable performance enhancement in automated pain detection when temporal information is included. Specifically, VST-0, which integrates temporal information, achieved an F1-score of 0.56 compared to 0.53 for ST-0 and an AUC of 0.85 compared to 0.80 for ST-0. This represents a significant improvement in both metrics by including temporal information.

The different performance between VST-0 and ST-0 clearly highlights the significance of the temporal component in automated pain detection. This comparative study was structured under nearly identical conditions for both models, including similar preprocessing steps, model size, and the number of layers, which strengthens the validity of our conclusion. These findings align with previous research on the temporal dynamics of pain (87)(88)(78)(67), which has demonstrated success in incorporating temporal information in non-transformer models. However, our results present a challenge to recent studies involving transformer models, such as the work by Fiorentini et al.(27), who did not observe significant improvements with the inclusion of temporal data in Vision Transformers compared to Video Vision

Transformers. In contrast to Fiorentini et al.'s method (27), which employed uniform frame sampling to create a 2x2 grid across frames in 2D format, our approach applies the tubelet embedding technique. This technique involves a more explicit integration of temporal depth with spatiotemporal 3D "tubes", suggesting that the method of incorporating temporal information is essential. In Fiorentini et al.'s study (27), the absence of a significant improvement might be attributed to their methodological approach rather than the ineffectiveness of temporal information itself. Our use of the tubelet technique, which aligns more closely with the inherent design principles of the VST, appears to be more effective for this application. This implies that not just the inclusion but also the method of integrating temporal data plays an essential role in enhancing the performance of transformer-based models in pain detection tasks.

The given insights from this sub-research question confirm the crucial role of temporal dynamics in improving the performance of automated pain detection systems, particularly for transformer-based models. Moreover, given the discovery that the method of integrating temporal data probably affects performance, further studies could compare different temporal integration techniques on several architectures to determine the most effective approach in automated pain assessment. In a more general context, the importance of temporal information could also be transferred to related tasks regarding facial expressions.

To what extent does increasing the temporal depth input of Video Swin Transformers enhance pain detection capabilities? To further understand the impact of temporal dynamics on pain detection capabilities, our research extended to exploring the effect of increasing the temporal depth in Video Swin Transformers. Building upon the positive findings with temporal integration, we compared the performance between VST models configured with four and eight frames. The extension to eight frames resulted in a significant performance improvement, with the F1-score increasing by 5.4% (from 0.56 to 0.59) and the AUC increasing by 2.4% (from 0.85 to 0.87).

Our results confirm that enhancing the temporal depth from four to eight frames not only retains the benefits of temporal integration but also improved them. While the prior research by Fiorentini et al.(27) have suggested that a minimum of four frames might be sufficient to capture the full dynamics of the most prolonged AU, our findings challenge this claim, demonstrating superior performance with an eight-frame temporal depth. This suggests that a four-frame depth may not adequately capture the complete dynamics of pain expressions.

Furthermore, our findings are in line with other related research, indicating that increased temporal depths can reach better performance in automated pain detection trained on UNBC McMaster. For instance, Rodriguez et al.'s application (67) of a 16-frame depth in their LSTM models, and Tavakolian and Hadid's research (78), which highlighted that a 32-frame temporal depth significantly outperformed models with a lower number of temporal depth, support our conclusions. The effect of a even higher temporal depth, for instance 16 or 32 frames, is not investigated in our research scope, but has potential for further investigations.

However, extending the temporal depth introduces a trade-off, particularly in terms of computational efficiency, which could impact the feasibility of real-time applications. The decline in computational efficiency with increased temporal depth necessitates a balanced approach, where the benefits of improved detection capabilities must be weighed against the increased computational demands. Further research is needed to identify an optimal balance, possibly through more detailed experiments that also consider computational time and resources used.

How does the use of Focal loss during training on the imbalanced UNBC McMaster dataset, in comparison with oversampling techniques, impact the detection of pain? This research question investigated the impact of using Focal loss versus the oversampling technique on the detection of pain in the imbalanced UNBC McMaster dataset. These included training the Video Swin Transformer model using Focal loss (VST-2-FL) and comparing its performance against the ones of the main VST using oversampling (VST-0). The comparative analysis gives insights into the efficiency of each method in addressing class imbalances and enhancing model performance. The findings showed that the VST-2-FL model trained with Focal loss reported an average F1-score of 0.44 and a mean AUC of 0.77, but significantly underperformed in all metrics compared to the VST-0 model. Furthermore, the VST-0 not only achieved higher scores but also demonstrated greater stability across different test folds.

These findings suggest that Focal loss might not be effective for addressing data imbalance in the context of automated pain detection. The reason could also include the limited experimental setup. For example, in the hyperparameter optimization of the Focal loss model, where essential hyperparameters such as the focusing parameter gamma and the weighting factor alpha were potentially not optimally set. As the number of trials for hyperparameter optimization was limited to 20 due to computational constraints, it may not have been sufficient to find the optimal settings. This

configuration might not provide the required emphasis on minority samples of the pain class, leading to suboptimal training results. Moreover, oversampling techniques, which increase the representation of minority classes by duplicating samples, might have provided a more straightforward and effective method for addressing the imbalance as it works in the data level. This approach directly changes the data distribution that the model encounters during training, which can sometimes achieve better performance than working on the classifier level during training. Another contributing factor could be the training constraint where models were trained for only one epoch due to early convergence observed in other models. This limited training period might not have allowed the Focal loss enough time to properly adjust and stabilize, which is particularly important given that Focal loss can sometimes lead to slower convergence or instability in training if not configured correctly.

Although this was the first known attempt to apply Focal loss to automated pain assessment, and the results did not favor Focal loss over the oversampling method, this exploration opens the door to further research. It highlights the need for more extensive hyperparameter tuning and potentially more adaptive training strategies to explore the full capabilities of Focal loss in imbalanced datasets like those used for pain detection. This initial study could motivate additional research to refine the application of Focal loss and investigate its efficiency with enhanced computational resources.

How do Video Swin Transformer-based pain detectors generalize across different pain contexts? This sub-research question explores the capability of our models, particularly our Video Swin Transformer, to generalize across diverse pain contexts by using cross-dataset validation. More specifically, our models trained on a clinical dataset, the UNBC McMaster, are tested on the BioVid dataset, which is an experimental dataset. The VST-0 model showed the highest average F1-score, while the ViT-0 model scored slightly better in terms of AUC and accuracy. Both model architectures indicate better generalizability across datasets compared to the ST-0 model, which consistently showed lower performance across all metrics. This pattern was also observed in the within-dataset validation, where the VST outperformed the Swin Transformer, strengthening the aspect of the spatiotemporal component also regarding generalizability.

Despite these insights, the results also present that none of the models has excellent generalization abilities, particularly when assessed on accuracy. Although the accuracy is above baseline, it still falls short of what might be considered good. This can be due to significant differences in the settings

of the datasets (clinical vs. experimental) or the presence (BioVid) versus absence (UNBC) of EEG caps, among other factors. Another difficulty in generalizing is the behavioral differences in the expression of pain captured across the datasets.

Our findings align with those of Prajod et al. (63), who noted similar challenges in generalization. They observed that models trained on the UNBC dataset struggled to recognize pain in the BioVid dataset due to differences in pain expression behavior. Particularly, participants in the BioVid dataset frequently closed their eyes, also in no pain cases, a behavior less common in the clinical context of the UNBC dataset. This behavioral deviation leads to significant performance decrease, as the models fail to generalize the specific pain-related characteristics from one context to another. Dai et al. (17) also highlight the difficulty of transferring learned pain recognition across datasets with distinct characteristics. They pointed out that the environment, as well as the experimental setup, such as the presence or absence of certain control equipment like EEG caps, can drastically affect the performance of pain assessment models. Ertugrul et al. (24), in their cross-domain experiments on AU detection, found similar challenges, indicating that generalizability issues go beyond automated pain detection and also affect AU detection models.

Our study helps fill the gap in cross-dataset validation in automated pain assessment research and contributes to the very few works that have tested their models in different pain context and dataset settings. Our findings examine the challenges of cross-dataset validation, particularly moving from a clinical to an experimental dataset, suggesting that further improvement in model generalizability is necessary. Furthermore, it highlights the need for models to be trained on more diverse datasets that include a broader range of pain expressions and contexts to improve their generalizability.

Can model-specific explainability methods generate plausible explanations for the outputs of Video Swin, Swin, Vision Transformer-based pain expression detection models, and how do the explanations generated differ among the model architectures? The last research questions is addressing the explainability of our models, particularly the plausibility of explanation and their differences between the models. Our study successfully applied the approach proposed by Nguyen et al. (61) to Swin Transformer models, which is a novelty in the field of automated pain assessment. This adaption, together with conventional ViT attention extraction method by Chefer et al. (13), allowed us to extract and analyze attention maps, showing how these models focus on their predictions.

Our models potentially learned to recognize pain-specific AUs effectively. This is indicated from their attention focusing mainly on regions associated with pain expressions. Common pain-related AUs identified across all models include AU4 (lowering of brows), AU7 (lids tight), and AU9 (nose wrinkling). In addition, for the VST and ViT, pain-specific AU43 (eyes closed), but also attention in the non-pain-specific chin region.

The relation to Fiorentini et al.'s results (27) further validates our approach, as similar attention patterns to pain-related areas have been demonstrated in their work, which reinforces the reliability of our models' learning focus.

Interestingly, we observed that background noise plays a role in the Swin Transformer's predictive errors, leading to false positives. This could demonstrate the disadvantages of using 2D face frontalization instead of its 3D version, even though its mentioned advantages.

This study is one of the first known work to apply the method by Nguyen et al. (61) for extracting attention maps from Swin Transformers. The success of these applications not only encourage for further investigations into the explainability of future models but also suggests potential uses of Swin Transformer explainability in other domains.

The implications of our findings extend to the broader field of explainable AI, particularly improving trustworthiness in automated systems used within the medical sector. By demonstrating that these models genuinely learn and focus on pain-related AUs, we contribute to the potential for these technologies to assist in clinical settings.

9.2 Limitations and future work

Model Comparison One of the limitations of our work is the differences in model size between the Video Swin Transformer and comparison models, specifically the ViT. As mentioned before, this needs to be considered and might lead to an unfair comparison on the architecture level. Including comparison models for a fairer comparison between ViT (base variant) and (Video) Swin Transformer (small variant) would require more computational availability to train them on base variants.

ViViT Comparison Not being able to include a ViViT comparison model to compare our VST directly on a spatiotemporal level with the Vision Transformer architecture is another limitation. This is due to the restricted choice of pre-trained models for ViViT and the limited computational resources to train it on existing pre-trained models. Future research can address this

problem and work on a comparable ViViT model that integrates temporal information through tubelet embeddings.

Temporal Depth Investigation While our findings suggest that increasing the temporal depth from four to eight frames improves performance, it remains unclear whether extending this further would continue to yield benefits. Future research should explore beyond the eight-frame setup, possibly testing up to 16 or 32 frames, to determine the optimal temporal depth that balances performance with computational efficiency.

Computational Aspects and Real-Time Applications While our research demonstrated the potential of the (Video) Swin Transformer architecture for automated pain detection, our focus was on effectiveness at "lower" resolutions (224x224). The (Video) Swin Transformer's advantage lies in its computational efficiency not correlating with resolution, making it superior for high-resolution images compared to Vision Transformers in terms of computational aspect. However, the computational aspect and the real-time applicability of (Video) Swin Transformer were not part of this research. This aspect can be further investigated to demonstrate the superiority of (Video) Swin Transformer in high-resolution cases in a comparative analysis with Vision Transformers, for example.

Hyperparameter Optimization Although hyperparameter optimization was performed in the experiment, it might be limited in terms of tested parameters and the number of trials. Due to the high computational costs associated with nested cross-validation in combination with the expense of our models, this research was restricted in finding the optimal hyperparameters. With more extensive hyperparameter optimization, the Focal loss models could be further improved and may get more insightful results.

Preprocessing and Background Noise Attention visualization indicated that background noise may affect model performance, particularly for the Swin Transformer model. This might be because of our preprocessing and the face frontalization on a 2D level instead of 3D. To avoid this, future work can investigate more in the 3D technique and try to mitigate the mentioned disadvantages of it.

Dataset Variability and Generalization Most of the current research, including this study, relies heavily on the UNBC McMaster dataset. Our cross-dataset validation showed, that there is a critical need for datasets

that covers a wider variety of pain contexts and expressions to ensure that models are not only effective but also generalizable across different settings and populations. Future work should focus on acquiring or creating more diverse datasets, which could help in developing models that are robust across various real-world scenarios.

Qualitative Evaluation Our results highlight that while certain AUs are recognized as pain-related across all models, such as the chin region, they are not universally pain-specific. This suggests a need for deeper exploration into the role and relevance of different AUs in pain detection. Future research could look into how models prioritize different AUs and the potential for discovering new pain-related AUs that have not yet been extensively studied.

Pain Intensity Estimation This study is limited to pain detection, but related tasks like pain intensity estimation or similar facial expression domains can be further researched. Future work can investigate these tasks using our method.

10 Conclusion

This research thesis investigated the capabilities of Video Swin Transformers in the automated assessment of pain through facial expressions. We compared the Video Swin Transformers models against both Swin Transformers and Vision Transformers to evaluate their performance efficiency.

Our findings demonstrated a notable advancement in the use of VST for pain detection. The main VST model, VST-0, showed promising results with an F1-score of 0.56±0.06 and an AUC of 0.85±0.04, which improved further with temporal depth optimization in the VST-1-TD model, achieving an F1-score of 0.59±0.04 and an AUC of 0.87±0.03. These results surpassed those of our ViT and Swin Transformer model, which highlights the benefit of incorporating temporal information and the potential of its architectural characteristics in automated pain assessment.

Moreover, our research addressed several other aspects. For example, the impact of increasing temporal depth, which showed improved detection capabilities, or the comparison of different techniques for handling imbalanced datasets and their effects on model performance. Furthermore, insights about generalizability and model explainability were given.

In conclusion, our Video Swin Transformer models have set new state-of-the-art performance in automated pain detection, offering improvements

over other transformer-based models, especially by effectively incorporating temporal dynamics. The architectural advantages of VSTs in handling high-resolution data and their computational efficiency make them a promising solution for real-time applications. The broader implications of this study not only enrich the field of automated pain detection but also set the way for future innovations in related areas of facial expression analysis.

References

[1] S. Abnar and W. Zuidema. Quantifying attention flow in transformers, 2020.

[2] W. Achterberg, M. Pieper, A. van Dalen-Kok, M. de Waal, B. Husebo, S. Lautenbacher, M. Kunz, E. Scherder, and A. Corbett. Pain management in patients with dementia. *Clinical Interventions in Aging*, 8:1471–1482, Nov 2013.

[3] T. Akiba, S. Sano, T. Yanase, T. Ohta, and M. Koyama. Optuna: A next-generation hyperparameter optimization framework. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2019.

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

[5] A. B. Ashraf, S. Lucey, J. F. Cohn, T. Chen, Z. Ambadar, K. Prkachin, P. Solomon, and B. J. Theobald. The painful face. *Proceedings of the 9th international conference on Multimodal interfaces*, 2007.

[6] S. Bach, A. Binder, G. Montavon, F. Klauschen, K.-R. Müller, and W. Samek. On pixel-wise explanations for non-linear classifier decisions by layer-wise relevance propagation. *PLOS ONE*, 10(7):1–46, 07 2015.

[7] A. Baevski and M. Auli. Adaptive input representations for neural language modeling, 2019.

[8] G. Bargshady, J. Soar, X. Zhou, R. Deo, F. Whittaker, and H. Wang. A joint deep neural network model for pain recognition from face. 02 2019.

[9] G. Bargshady, X. Zhou, R. Deo, J. Soar, F. Whittaker, and H. Wang. Enhanced deep learning algorithm development to detect pain intensity

from facial expression images. *Expert Systems with Applications*, 149:1–10, July 2020.

[10] E. E. Benarroch. Pain-autonomic interactions: A selective review. *Clinical Autonomic Research*, 11(6):343–349, 2001.

[11] M. Buda, A. Maki, and M. A. Mazurowski. A systematic study of the class imbalance problem in convolutional neural networks. *Neural Networks*, 106:249–259, Oct. 2018.

[12] H. Chefer, S. Gur, and L. Wolf. Generic attention-model explainability for interpreting bi-modal and encoder-decoder transformers, 2021.

[13] H. Chefer, S. Gur, and L. Wolf. Transformer interpretability beyond attention visualization, 2021.

[14] G. A. S. Coutrin, L. P. Carlini, L. A. Ferreira, T. M. Heiderich, R. C. X. Balda, M. C. M. Barros, R. Guinsburg, and C. E. Thomaz. Convolutional neural networks for newborn pain assessment using face images: A quantitative and qualitative comparison. In R. Su, Y. Zhang, H. Liu, and A. F Frangi, editors, *Medical Imaging and Computer-Aided Diagnosis*, pages 503–513, Singapore, 2023. Springer Nature Singapore.

[15] K. D. Craig. The facial expression of pain better than a thousand words? *APS Journal*, 1(3):153–162, 1992.

[16] K. D. Craig. Social communication model of pain. *Pain*, 156(7):1198–1199, 2015.

[17] L. Dai, J. Broekens, and K. P. Truong. Real-time pain detection in facial expressions for health robotics. In *2019 8th International Conference on Affective Computing and Intelligent Interaction Workshops and Demos (ACIIW)*, pages 277–283, 2019.

[18] R. de Wit, F. van Dam, M. Hanneman, L. Zandbelt, A. van Buuren, K. van der Heijden, G. Leenhouts, S. Loonstra, and H. H. Abu-Saad. Evaluation of the use of a pain diary in chronic cancer pain patients at home. *Pain*, 79(1):89–99, 1999.

[19] J. Deng, J. Guo, Y. Zhou, J. Yu, I. Kotsia, and S. Zafeiriou. Retinaface: Single-stage dense face localisation in the wild, 2019.

[20] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly,

J. Uszkoreit, and N. Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale, 2021.

[21] W. Downie, P. Leatham, V. Rhind, V. Wright, J. Branco, and J. Anderson. Studies with pain rating scales. *Annals of the Rheumatic Diseases*, 37(4):378–381, Aug 1978.

[22] P. Ekman and W. V. Friesen. Facial action coding system. *PsycTESTS Dataset*, 1978.

[23] V. F. Engle, M. J. Graney, and A. Chan. Accuracy and bias of licensed practical nurse and nursing assistant ratings of nursing home residents' pain. *The Journals of Gerontology Series A: Biological Sciences and Medical Sciences*, 56(7), 2001.

[24] I. O. Ertugrul, J. F. Cohn, L. A. Jeni, Z. Zhang, L. Yin, and Q. Ji. Cross-domain au detection: Domains, learning approaches, and measures. In *2019 14th IEEE International Conference on Automatic Face Gesture Recognition (FG 2019)*, pages 1–8, 2019.

[25] H. M. et al. Pain terms: a list with definitions and notes on usage. recommended by the iasp subcommittee on taxonomy., 06 1979.

[26] Y. Feng, F. Wu, X. Shao, Y. Wang, and X. Zhou. Joint 3d face reconstruction and dense alignment with position map regression network, 2018.

[27] G. Fiorentini, I. O. Ertugrul, and A. A. Salah. Fully-attentive and interpretable: vision and video vision transformers for pain detection, 2022.

[28] R. Fong, M. Patrick, and A. Vedaldi. Understanding deep networks via extremal perturbations and smooth masks, 2019.

[29] R. C. Fong and A. Vedaldi. Interpretable explanations of black boxes by meaningful perturbation. In *2017 IEEE International Conference on Computer Vision (ICCV)*. IEEE, Oct. 2017.

[30] A. Gawande. The checklist manifesto: How to get things right. *Journal of Nursing Regulation*, 1(4):64, 2011.

[31] S. Gkikas and M. Tsiknakis. Automatic assessment of pain based on deep learning methods: A systematic review. *Computer Methods and Programs in Biomedicine*, 231:107365, 2023.

[32] E. He, Q. Chen, and Q. Zhong. Sl-swin: A transformer-based deep learning approach for macro- and micro-expression spotting on small-size expression datasets. *Electronics*, 12(12), 2023.

[33] K. Herr, P. J. Coyne, M. McCaffery, R. Manworren, and S. Merkel. Pain assessment in the patient unable to self-report: Position statement with clinical practice recommendations. *Pain Management Nursing*, 12(4):230–250, 2011.

[34] D. Huang, Z. Xia, J. Mwesigye, and X. Feng. Pain-attentive network: a deep spatio-temporal attention model for pain estimation. *Multimedia Tools and Applications*, 79:1–26, 10 2020.

[35] L. Jeni, J. Cohn, and F. De la Torre. Facing imbalanced data - recommendations for the use of performance metrics. volume 2013, 09 2013.

[36] L. Jeni, J. Cohn, and F. De la Torre. Facing imbalanced data - recommendations for the use of performance metrics. volume 2013, 09 2013.

[37] J. Kappesser and A. C. de C. Williams. Pain estimation: Asking the right questions. *Pain*, 148(2):184–187, 2010.

[38] W. Kay, J. Carreira, K. Simonyan, B. Zhang, C. Hillier, S. Vijayanarasimhan, F. Viola, T. Green, T. Back, P. Natsev, M. Suleyman, and A. Zisserman. The kinetics human action video dataset, 2017.

[39] R. A. Khan, A. Meyer, H. Konik, and S. Bouakaz. Pain detection through shape and appearance features. *2013 IEEE International Conference on Multimedia and Expo (ICME)*, 2013.

[40] B. Kim, R. Khanna, and O. O. Koyejo. Examples are not enough, learn to criticize! criticism for interpretability. In D. Lee, M. Sugiyama, U. Luxburg, I. Guyon, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 29. Curran Associates, Inc., 2016.

[41] J.-H. Kim, N. Kim, and C. S. Won. Facial expression recognition with swin transformer. *ArXiv*, abs/2203.13472, 2022.

[42] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization, 2017.

[43] M. Kunz, S. Scharmann, U. Hemmeter, K. Schepelmann, and S. Lautenbacher. The facial expression of pain in patients with dementia. *Pain*, 133(1):221–228, 2007.

[44] S. A. Lakshminarayan, S. Hinduja, and S. Canavan. Three-level training of multi-head architecture for pain detection. *2020 15th IEEE International Conference on Automatic Face and Gesture Recognition (FG 2020)*, 2020.

[45] D. N. T. Le, H. X. Le, L. T. Ngo, and H. T. Ngo. Transfer learning with class-weighted and focal loss function for automatic skin cancer classification, 2020.

[46] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár. Focal loss for dense object detection, 2018.

[47] Z. Liu, Y. Lin, Y. Cao, H. Hu, Y. Wei, Z. Zhang, S. Lin, and B. Guo. Swin transformer: Hierarchical vision transformer using shifted windows, 2021.

[48] Z. Liu, J. Ning, Y. Cao, Y. Wei, Z. Zhang, S. Lin, and H. Hu. Video swin transformer, 2021.

[49] P. Lucey, J. F. Cohn, K. M. Prkachin, P. E. Solomon, and I. Matthews. Painful data: The unbc-mcmaster shoulder pain expression archive database. In *2011 IEEE International Conference on Automatic Face amp; Gesture Recognition (FG 2011)*, pages 57–64, Los Alamitos, CA, USA, March 2011. IEEE Computer Society.

[50] P. Lucey, J. F. Cohn, K. M. Prkachin, P. E. Solomon, and I. Matthews. Painful data: The unbc-mcmaster shoulder pain expression archive database. In *2011 IEEE International Conference on Automatic Face Gesture Recognition (FG)*, pages 57–64, 2011.

[51] S. Lundberg and S.-I. Lee. A unified approach to interpreting model predictions, 2017.

[52] A. Majumder, L. Behera, and V. Subramanian. Gmr based pain intensity recognition using imbalanced data handling techniques. pages 1–5, 10 2016.

[53] D. L. Martinez, O. Rudovic, and R. W. Picard. Personalized automatic estimation of self-reported pain intensity from facial expressions. *CoRR*, abs/1706.07154, 2017.

[54] H. M. McCormack, D. J. de L. Horne, and S. Sheather. Clinical applications of visual analogue scales: A critical review. *Psychological Medicine*, 18(4):1007–1019, 1988.

[55] Q. McNemar. Note on the sampling error of the difference between correlated proportions or percentages. *Psychometrika*, 12(2):153–157, June 1947.

[56] R. Melzack and J. D. Loeser. Pain - an overview. *Acta Anaesthesiologica Scandinavica*, 43(9):880–884, 1999.

[57] T. Miller. Explanation in artificial intelligence: Insights from the social sciences, 2018.

[58] R. Mohammed, J. Rawashdeh, and M. Abdullah. Machine learning with oversampling and undersampling techniques: Overview study and experimental results. pages 243–248, 04 2020.

[59] S. Nerella, K. Khezeli, A. Davidson, P. Tighe, A. Bihorac, and P. Rashidi. End-to-end machine learning framework for facial au detection in intensive care units, 2022.

[60] N. Neshov and A. Manolova. Pain detection from facial characteristics using supervised descent method. *2015 IEEE 8th International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications (IDAACS)*, 2015.

[61] H. C. Nguyen, H. Lee, and J. Kim. Inspecting explainability of transformer models with additional statistical information, 2023.

[62] E. Othman, P. Werner, F. Saxen, A. Al-Hamadi, and S. Walter. Cross-database evaluation of pain recognition from facial video. In *2019 11th International Symposium on Image and Signal Processing and Analysis (ISPA)*, pages 181–186, 2019.

[63] P. Prajod, T. Huber, and E. André. Using explainable ai to identify differences between clinical and experimental pain detection models based on facial expressions. In B. ór Jónsson, C. Gurrin, M.-T. Tran, D.-T. Dang-Nguyen, A. M.-C. Hu, B. Huynh Thi Thanh, and B. Huet, editors, *MultiMedia Modeling*. Springer International Publishing, 2022.

[64] K. M. Prkachin. The consistency of facial expressions of pain: A comparison across modalities. *Pain*, 51(3):297–306, 1992.

[65] K. M. Prkachin and P. E. Solomon. The structure, reliability and validity of pain expression: Evidence from patients with shoulder pain. *Pain*, 139(2):267–274, 2008.

[66] S. N. Raja, D. B. Carr, M. Cohen, N. B. Finnerup, H. Flor, S. Gibson, F. J. Keefe, J. S. Mogil, M. Ringkamp, K. A. Sluka, and et al. The revised international association for the study of pain definition of pain: Concepts, challenges, and compromises. *Pain*, 161(9):1976–1982, 2020.

[67] P. Rodriguez, G. Cucurull, J. Gonzalez, J. M. Gonfaus, K. Nasrollahi, T. B. Moeslund, and F. X. Roca. Deep pain: Exploiting long short-term memory networks for facial expression classification. *IEEE Transactions on Cybernetics*, 52(5):3314–3324, 2022.

[68] S. Ruder. An overview of gradient descent optimization algorithms, 2017.

[69] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei. ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision (IJCV)*, 115(3):211–252, 2015.

[70] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra. Grad-cam: Visual explanations from deep networks via gradient-based localization. *International Journal of Computer Vision*, 128(2):336–359, Oct. 2019.

[71] A. Semwal and N. D. Londhe. Automated pain severity detection using convolutional neural network. *2018 International Conference on Computational Techniques, Electronics and Mechanical Systems (CTEMS)*, 2018.

[72] A. Semwal and N. D. Londhe. Computer aided pain detection and intensity estimation using compact cnn based fusion network. *Applied Soft Computing*, 112:107780, 2021.

[73] S. I. Serengil and A. Ozpinar. Lightface: A hybrid deep face recognition framework. In *2020 Innovations in Intelligent Systems and Applications Conference (ASYU)*, pages 23–27. IEEE, 2020.

[74] S. I. Serengil and A. Ozpinar. Hyperextended lightface: A facial attribute analysis framework. In *2021 International Conference on Engineering and Emerging Technologies (ICEET)*, pages 1–4. IEEE, 2021.

[75] E. Sheu, J. Versloot, R. Nader, D. Kerr, and K. D. Craig. Pain in the elderly. *The Clinical Journal of Pain*, 27(7):593–601, 2011.

[76] D. Simon, K. D. Craig, F. Gosselin, P. Belin, and P. Rainville. Recognition and discrimination of prototypical dynamic expressions of pain and emotions. *Pain*, 135(1):55–64, 2008.

[77] Z. Sun and G. Tzimiropoulos. Part-based face recognition with vision transformers, 2022.

[78] M. Tavakolian and A. Hadid. A spatiotemporal convolutional neural network for automatic pain intensity estimation from facial dynamics. *International Journal of Computer Vision*, 127(10):1413–1425, 2019.

[79] G. S. Tran, T. P. Nghiem, V. T. Nguyen, C. M. Luong, and J.-C. Burie. Improving accuracy of lung nodule classification using deep learning with focal loss. *Journal of Healthcare Engineering*, 2019, 2019.

[80] D. C. Turk, R. Melzack, K. D. Craig, K. M. Prkachin, and R. E. Grunau. *The facial expression of pain*, page 153–169. Guilford Press, 2011.

[81] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin. Attention is all you need. 2023.

[82] T. Vu, V. T. Huynh, and S. H. Kim. Vision transformer for action units detection, 2023.

[83] S. Walter, S. Gruss, H. Ehleiter, J. Tan, H. C. Traue, P. Werner, A. Al-Hamadi, S. Crawcour, A. O. Andrade, and G. Moreira da Silva. The biovid heat pain database: data for the advancement and systematic validation of an automated pain recognition system. In *2013 IEEE International Conference on Cybernetics (CYBCO)*, pages 128–131, 2013.

[84] C. Wang and Z. Wang. Progressive multi-scale vision transformer for facial action unit detection. *Frontiers in Neurorobotics*, 15, 01 2022.

[85] Q. Wang, B. Li, T. Xiao, J. Zhu, C. Li, D. F. Wong, and L. S. Chao. Learning deep transformer models for machine translation. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1810–1822, Florence, Italy, July 2019. Association for Computational Linguistics.

[86] K. Weitz, T. Hassan, U. Schmid, and J.-U. Garbas. Deep-learned faces of pain and emotions: Elucidating the differences of facial expressions with the help of explainable ai methods. *tm - Technisches Messen*, 86(7-8):404–412, 2019.

[87] P. Werner, A. Al-Hamadi, K. Limbrecht-Ecklundt, S. Walter, S. Gruss, and H. C. Traue. Automatic pain assessment with facial activity descriptors. *IEEE Transactions on Affective Computing*, 8(3):286–299, 2017.

[88] P. Werner, A. Al-Hamadi, and S. Walter. Analysis of facial expressiveness during experimentally induced heat pain. *2017 Seventh International Conference on Affective Computing and Intelligent Interaction Workshops and Demos (ACIIW)*, 2017.

[89] P. Werner, D. Lopez-Martinez, S. Walter, A. Al-Hamadi, S. Gruss, and R. W. Picard. Automatic recognition methods supporting pain assessment: A survey. *IEEE Transactions on Affective Computing*, 13(1):530–552, 2022.

[90] A. C. Williams. Facial expression of pain: An evolutionary account. *Behavioral and Brain Sciences*, 25(04), 2002.

[91] H. Xu and M. Liu. A deep attention transformer network for pain estimation with facial expression video. In J. Feng, J. Zhang, M. Liu, and Y. Fang, editors, *Biometric Recognition*, pages 112–119, Cham, 2021. Springer International Publishing.

[92] R. Yang, X. Hong, J. Peng, X. Feng, and G. Zhao. Incorporating high-level and low-level cues for pain intensity estimation. *2018 24th International Conference on Pattern Recognition (ICPR)*, 2018.

[93] R. Yang, S. Tong, M. Bordallo Lopez, E. Boutellaa, J. Peng, X. Feng, and A. Hadid. On pain assessment from facial videos using spatio-temporal local descriptors. 12 2016.

[94] X. Yuan, S. Zhang, C. Zhao, X. He, B. Ouyang, and S. Yang. Pain intensity recognition from masked facial expressions using swin-transformer. In *2022 IEEE International Conference on Robotics and Biomimetics (ROBIO)*, pages 723–728, 2022.

[95] J. Zhang, W. Li, and P. Ogunbona. Transfer learning for cross-dataset recognition: A survey. 05 2017.

[96] S. M. Zwakhalen, J. P. Hamers, H. H. Abu-Saad, and M. P. Berger. Pain in elderly people with severe dementia: A systematic review of behavioural pain assessment tools. *BMC Geriatrics*, 6(1), 2006.