February 2021

Pain Recognition Performance on a Single Board Computer

Iyonna L. Tynes
University of South Florida

Pain Recognition Performance on a Single Board Computer

by

Iyonna L. Tynes

A thesis submitted in partial fulfillment
of the requirements for the degree of
Master of Science
Department of Computer Science and Engineering
College of Engineering
University of South Florida

Major Professor: Shaun Canavan, Ph.D.
Robert Karam, Ph.D.
Tempestt Neal, Ph.D.

Date of Approval:
March 19, 2021

Keywords: emotion recognition, machine learning, Jetson Nano, Raspberry Pi, ONNX

Table of Contents

List of Tables

List of Figures

Abstract

Emotion recognition is a quickly growing field of study due to the increased interest in building systems which can classify and respond to emotions. Recent medical crises, such as the opioid overdose epidemic in the United States and the global COVID-19 pandemic has emphasized the importance of emotion recognition applications is areas like Telehealth services. Considering this, this thesis focuses specifically on pain recognition. The problem of pain recognition is approached from both a hardware and software perspective, as we propose a real-time pain recognition system, from facial images, that is deployed on an NVIDIA Jetson Nano single-board computer. We have conducted offline experiments using the BP4D dataset, where we investigate the impact of gender and data imbalance. This thesis proposes an affordable and easily accessible system which could perform pain recognition inferences. The results from this study found a balanced dataset, in terms of class and gender, results in the highest accuracies for pain recognition. We also detail the difficulties of pain recognition using facial images and propose some future work that can be investigated for this challenging problem.

Chapter 1: Introduction

1.1 Motivation and Problem Statement

Continued interest in improving human-computer interactions (HCI) has led to the field of affective computing [17], which is the ability to correctly identify and appropriately respond to human emotions in devices such as PCs, smart phones, and other ubiquitous devices [11]. The utility of this field is diverse, and its scope is widespread across a multitude of sectors such as market analytics [1], healthcare [12], and entertainment [9]. Automotive companies Audi, Mercedes, and Volvo have already integrated drowsiness detection safety features which, in addition to monitoring automobile behavior including lane position and steering wheel movement, also analyze human metrics such as gaze position and eye-blink rate [6].

Prior to the year 2020, America was confronting an opioid overdose epidemic. In 2019 an estimated ten million Americans were misusing opioids and approximately 128 daily deaths were the result of an opioid overdose [5], [27]. Hospitals became both a facilitator and a victim of the crisis. It has been shown that some clinicians believe emergency departments in hospitals had a habit of inappropriately administering opioids to patients and this behavior had contributed to the opioid epidemic [12]. As a result, hospitals have had to train healthcare workers to identify and report potential “drug-seekers”, or individuals who seek out medical prescriptions to satiate an addiction rather than for the intended use of pain management [15]. In recent years, devices which could successfully implement emotion recognition, and more specifically pain recognition, were considered as a tool to assist physicians in discerning genuine physical distress in patients. The Covid-19 pandemic has further revealed the practicality of pain recognition technologies. Requests

for Telehealth services increased 3.06% between October 2019 and October 2020, increasing from 0.18% to 5.61% [19]. There is also evidence to support the permanence of Telehealth services post-pandemic [25].

This thesis further investigates the viability of pain recognition, using facial images, from both a hardware and software perspective. Convolutional neural networks (CNNs) were developed for the proposed system, that can be deployed in a real-time setting. We evaluated a range of CNNs with different numbers of layers, as well as parameters. We evaluated the efficacy of the CNNs to recognize pain by evaluating the BP4D multimodal emotion dataset [29], [30]. Along with this, we also investigated the impact of data imbalance and gender on the accuracies of pain recognition. The best performing model was then formatted and deployed to a Jetson Nano single-board computer which had a Raspberry Pi camera module attached, for real-time recognition of pain.

The rest of this thesis is organized as follows. Chapter 1.2 details the specific contributions of this thesis and Chapter 2 includes a survey of previous research done in the areas of emotion and pain recognition. A description of the dataset used to train and test (offline) the CNN follows along with the preprocessing methods applied to the images are detailed in Chapter 3. Chapter 4 details the proposed experimental design and results, and Chapter 5 details the proposed real-time pain recognition system that was deployed on an NVIDIA Jetson Nano single-board computer. Chapter 6 finishes this thesis with conclusions, limitations, future challenges, and work to address these challenges.

1.2 Contributions

The main contribution of this work is an investigation into pain recognition from facial expressions. The specific contributions of this thesis are detailed below.

1. Analysis of automatically detecting pain using facial expressions. The study shows encouraging results for pain detection, while also detailing the difficulties of these types of automated systems.

2. Investigation into the impact of gender and class imbalance on pain detection accuracy. This study shows both gender and class imbalance can have a negative impact on pain detection accuracy.

3. Real-time implementation of proposed approach on NVIDIA Jetson Nano single-board computer. This study shows the proposed approach can be deployed to real-world environments with relatively affordable and easily setup equipment, paving the way for future investigation into real-time pain detection.

Chapter 2: Related Works

Recent research in emotion recognition has sought to improve the practice in a multitude of ways. One approach is to use visual inputs (e.g. images) and increase model performance by exploring different methods of feature extraction [10]. Another approach is to derive emotion classification on input types other than visual, such as physiological signals like the EEG, or electroencephalogram [8][28][31]. In this Chapter, we detail some works related to general emotion recognition, as well as pain recognition.

Bargal et al. [23] outlined a novel feature extraction approach used to improve emotion recognition using only visual inputs. In the study three different models were developed; two were derivatives of the VGG network and the final model was based on the RESNET network. Features were extracted from two fully connected layers, one from each of the VGG networks, and one pooling layer included in the RESNET model. The three features were ordered in various combinations to determine the best vector for training and classification. The leading feature vector achieved a validation accuracy of 59.52% and a testing accuracy of 56.66%. These results were a significant improvement over the state-of-the-art accuracies of 38.81% and 40.47% for validation and testing, respectively. Many attempts to address the problem of emotion recognition primarily utilize input images where the face is displayed in full-frontal view [26][33]. Considering this, Zheng et al. [34] proposed a novel approach to strengthen emotion recognition against pose variance. The proposed method represents images using the region covariance matrix (RCM). The RCM removes the need to align the face or distribute features locally in a facial region. The authors of this study tested their method against images with various pitch and yaw angles and measured

the error rate percentages for various emotions. The best average error rate of 25.17% was achieved when the pitch angle and yaw angles were both zero degrees. The error rates increased proportionally to the magnitude of change in pitch and yaw angles from zero degrees.

In a departure from the works described above, Wang et al. [31] used EEG signals as inputs to classify emotional states. Physiological data is preferable to visual physical data because it is controlled by the autonomic nervous system (ANS) and is resistant to manipulation. Outward expressions of emotions can be changed or distorted to hide the true emotional state. This lack of transparency in emotional responses presents a significant challenge to improving emotion recognition. Using EEG data, Wang et al. were able to achieve an accuracy of 91.77%.

While the above works focused on general emotion recognition (e.g. happy, sad, etc.), there has been promising research into pain recognition. An early study showing the feasibility of automatic pain recognition was from Lucey et al. [13]. They showed pain can be recognized using facial actions. An active appearance model [3] is used to track the face, which is then used to extract features for training a support vector machine. Through the detection of action units, they showed that different parts of the face can result in different accuracies for pain recognition. More recently, Hinduja et al. [10] showed that by fusing physiological signals with facial action units, pain recognition accuracy can be improved. They also showed there is a correlation between the physiological signals and the action units during the most expressive parts of a sequence [33]. Uddin et al. [26] showed that affect, other than pain, needs to be accounted for real-world pain recognition scenarios. They showed that when affect including, but not limited to, happy and sad were added to the training and testing data the accuracy decreased compared to only baseline and pain. As other affect can occur during pain, their study suggests that without accounting for other affect in pain recognition systems, the results of such as system may not be accurate.

Chapter 3: Dataset and Preprocessing

3.1 BP4D-Spontaneous Dataset

The Binghamton-Pittsburgh 3D Dynamic (4D) Spontaneous Facial Expression Database (BP4D-Spontaneous) [29], [30] was used to validate the efficacy of the proposed approach for pain recognition. The BP4D-Spontaneous dataset is a collection of 2D images, 3D features, Facial Action Coding System (FACS) [7] annotations (Action Units-AUs), captured from each participant. The dataset contains data from forty-one different participants (twenty-three female and eighteen male) who ranged in age from eighteen to twenty-nine years. The participants were subjected to various tasks which were designed to specifically elicit one of eight emotional responses: happiness, sadness, surprise, embarrassment, fear, pain, anger, or disgust. See Table 1 for the task number, task completed and the expected, elicited emotion. It is important to note, that these are the expected elicited emotion from the task. This is not the subject self-report of the emotion. To conduct our experiments, we used a subset of this dataset, which is detailed next in Subsection 3.2.

3.2 BP4D-Spontaneous Subset Dataset

To create this study's specific dataset subset thirty-six participants were randomly chosen from the original forty-one and the sampling was split evenly between male and female participants (eighteen female and eighteen male). For each participant, 420 total frames were chosen, which were split across the eight emotions. The frames chosen were empirically determined to be the best representations of the respective emotions (e.g. happy) for which the frames were collected. See Figure 1 for a sample of the participants and emotions included in the

subset dataset. Of the 420 frames, 210 were labeled as ‘pain’ and were selected from the sixth task in the study which prompted the pain response. The remaining 210 frames where split evenly among the other seven emotions (happiness, sadness, surprise, embarrassment, fear, pain, anger, disgust) and were collectively labeled as ‘no pain’. The frames remained as color images but were resized to 128 pixels by 128 pixels to fit the dimensions of the experimental convolutional neural network input layer, which is detailed in Chapter 4.

Table 1 BP4D-Spontaneous Dataset [29][30] Experimental Tasks and Expected Emotion

Task Number	Task Completed	Expected Elicited Emotion
1	Conversing with the experimenter	Happiness
2	Watching and reacting to a recorded documentary	Sadness
3	Experiencing a sudden noise	Surprise
4	Improvising of a silly song	Embarrassment
5	Perceiving a physical threat	Fear
6	Submerging hand in ice cold water	Pain
7	Receiving insults from the experimenter	Anger
8	Being exposed to an unpleasant odor.	Disgust

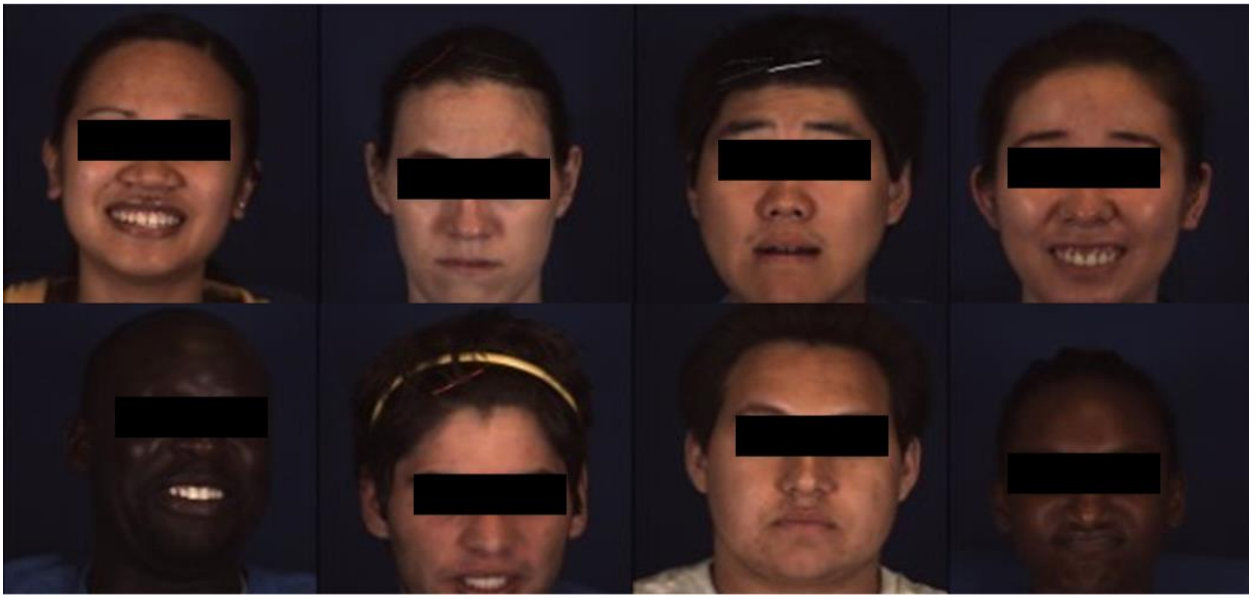

Figure 1 BP4D dataset [29][30] subset sample of featured participants and emotions. Emotions shown in top row, left to right: happy, sad, surprised, embarrassed. Emotions shown in bottom row, left to right: fear, pain, anger, disgust.

Chapter 4: Experimental Design and Results

The experimental pain detection CNN model was built using Keras and TensorFlow in a Python environment. CNN's architectures are a combination of convolutional layers, subsampling layers, and fully connected layers [22]. CNNs receive 2D array inputs and learn features by applying a sliding filter of size n x n to the image, which produces a smaller array of convolved features [22]. The experimental pain detection model has the input layer, three convolutional layers, two fully connected layers, and the output layer. The input layer accepts arrays of size (128 x 128 x 3). The first convolutional layer contains sixteen nodes and has a kernel size of (3 x 3). The following convolutional layer contains thirty-two nodes and has a kernel size of (3 x 3). The final convolutional layer contains sixty-four nodes and has a kernel size of (3 x 3). The first and second fully connected layers contain sixty-four and sixteen nodes, respectively. The output layer is sigmoid activated and has two outputs. See Figure 2 for a diagram of the proposed architecture.

Five experiments were conducted using the crafted BP4D dataset subset (Chapter 3.2) and the CNN model architecture described above. The goal of these experiments was to establish a relationship between dataset characteristics and produce the highest performing pain recognition model. The experiments involved a k-fold cross-validation study of the entire dataset subset, two gendered studies in which the two genders were split among training and testing, two gendered k-fold cross-validation studies, and manipulating the dataset subset with different distributions of the 'pain' class and the 'no pain' class.

4.1 Experiment 1: 9-Fold Cross-Validation Study

The first experiment focused on a k-fold cross validation study of the entire BP4D dataset subset. The thirty-six participants included in the subset were randomly sorted into nine groups, or folds. There were nine rounds of testing in which each fold was used as the test set in training once. Three models were built for each of the nine rounds and the model's validation loss and validation accuracy were recorded upon completion. The training sets for the 9-fold cross validation study contained 13,440 images and the testing sets contained 1,680 images. The training sets and the testing sets were both split evenly between the 'pain' and 'no pain' classes. The results from the study are listed in Table 2.

4.2 Experiment 2: Gendered Training and Testing

Once the 9-fold cross-validation study was complete, the thirty-six participants in the BP4D dataset subset were divided into two groups based on gender (male and female). Two rounds of testing followed in which one gender comprised the entirety of the training set, while the other encompassed the testing set. For each of the two rounds of testing the eighteen participants were found in the training set for a total of 7,560 frames. Three participants were randomly chosen from the opposite gender to build the training set for a total of 1,260 frames. Again, training sets and testing sets were split evenly between the two classes. Three models were built from each round of testing. The results from the experiment are listed in Table 3.

4.3 Experiment 3: Male 6-Fold Cross-Validation Study

The third experiment involved a k-fold cross-validation study on the male participants in the BP4D dataset subset. The eighteen male participants were split into six folds of three male participants each. Like the 9-fold cross-validation study conducted earlier; six rounds of testing occurred with each fold designated as the test set once. The training sets for the six-fold cross-

validation study had 8,820 images and the testing sets had 1,260 images. Both the training sets and testing sets were split evenly between the two classes. Three models were built for each round of testing. Table 4 shows the results for cross-validation experimentation.

4.4 Experiment 4: Female 6-Fold Cross-Validation Study

The female 6-fold cross-validation study follows the exact same structure as the male 6-fold cross-validation study outlined in the previous section. Eighteen participants were split evenly into six folds. The cross-validation study consisted of six rounds, with a new fold acting as the training set. The training set had 8,820 images and the testing set had 1,260 images with equal representation from both classes. Three models were built for each round of testing. The results from the experiment are listed in Table 5.

4.5 Experiment 5: Class Distribution Study

The final experiment focused on modifying the distribution of the training sets in one of two areas: gender and classification. All the previous four experiments divided training sets evenly between the two genders, where applicable, as well as having equal contributions between the two classes. The first part of this experiment changed the ratio between genders in the training sets to 60:40. The number of images in the training sets totaled 12,600 with 7,560 images coming from one gender and the remaining 5,040 from the opposite gender. The testing set contained 1,680 images compiled from two male participants and two female participants. The class contribution remained an even split in both the training sets and the testing sets for this portion of the experiment.

The second part of this experiment changed the ratio between classes in the training sets to 60:40. Likewise, there were 12,600 images in the training sets. One class contributed 7,560, while the opposing class contributed 5,040. Again, two male participants and two female participants

comprised the testing set, which equaled 1,680 images. The training sets and the testing sets had equal representation from both genders in this portion of the experiment. The results from this experiment are listed in Table 6.

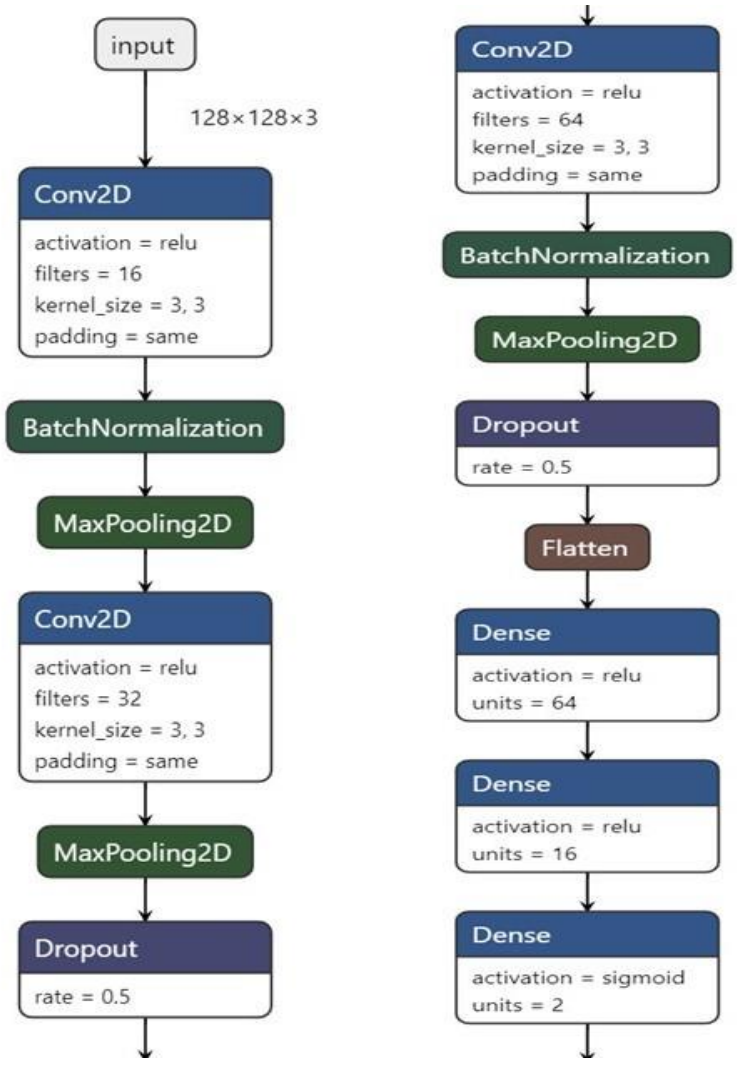

Figure 2 Experimental pain recognition model architecture. Visualization created using the Netron web Application [20]

Table 2 BP4D-Spontaneous Dataset 9-Fold Testing

	Build 1	Build 2	Build 3	Average
Fold 1	Validation Loss: 0.810 Validation Accuracy: 0.682	Validation Loss: 1.061 Validation Accuracy: 0.653	Validation Loss: 0.750 Validation Accuracy: 0.338	Validation Loss: 0.873 Validation Accuracy: 0.558
Fold 2	Validation Loss: 0.694 Validation Accuracy: 0.808	Validation Loss: 1.146 Validation Accuracy: 0.693	Validation Loss: 1.217 Validation Accuracy: 0.682	Validation Loss: 1.019 Validation Accuracy: 0.727
Fold 3	Validation Loss: 1.024 Validation Accuracy: 0.792	Validation Loss: 1.141 Validation Accuracy: 0.445	Validation Loss: 0.890 Validation Accuracy: 0.854	Validation Loss: 1.018 Validation Accuracy: 0.697
Fold 4	Validation Loss: 1.832 Validation Accuracy: 0.356	Validation Loss: 2.671 Validation Accuracy: 0.420	Validation Loss: 2.444 Validation Accuracy: 0.188	Validation Loss: 2.315 Validation Accuracy: 0.321
Fold 5	Validation Loss: 2.119 Validation Accuracy: 0.620	Validation Loss: 0.757 Validation Accuracy: 0.455	Validation Loss: 2.704 Validation Accuracy: 0.627	Validation Loss: 1.860 Validation Accuracy: 0.567
Fold 6	Validation Loss: 3.313 Validation Accuracy: 0.137	Validation Loss: 4.102 Validation Accuracy: 0.899	Validation Loss: 2.582 Validation Accuracy: 0.489	Validation Loss: 3.332 Validation Accuracy: 0.508
Fold 7	Validation Loss: 0.9664 Validation Accuracy: 0.5655	Validation Loss: 1.4322 Validation Accuracy: 0.1369	Validation Loss: 2.0304 Validation Accuracy: 0.873	Validation Loss: 1.476 Validation Accuracy: 0.525
Fold 8	Validation Loss: 1.2708 Validation Accuracy: 0.7937	Validation Loss: 1.6078 Validation Accuracy: 0.4167	Validation Loss: 0.5168 Validation Accuracy: 0.5516	Validation Loss: 1.132 Validation Accuracy: 0.587
Fold 9	Validation Loss: 0.8815 Validation Accuracy: 0.1657	Validation Loss: 2.1536 Validation Accuracy: 0.6032	Validation Loss: 0.9881 Validation Accuracy: 0.4524	Validation Loss: 1.341 Validation Accuracy: 0.407

Table 3 BP4D-Spontaneous Dataset Gender Study

	Build 1	Build 2	Build 3	Average
Male Train / Female Test	Validation Loss: 2.701 Validation Accuracy: 0.655	Validation Loss: 1.465 Validation Accuracy: 0.668	Validation Loss: 1.681 Validation Accuracy: 0.524	Validation Loss: 1.949 Validation Accuracy: 0.616
Female Train / Male Test	Validation Loss: 2.296 Validation Accuracy: 0.460	Validation Loss: 1.957 Validation Accuracy: 0.476	Validation Loss: 2.853 Validation Accuracy: 0.669	Validation Loss: 2.369 Validation Accuracy: 0.535

Table 4 BP4D-Spontaneous Dataset Male 6-Fold Testing

	Build 1	Build 2	Build 3	Average
Fold 1	Validation Loss: 3.724 Validation Accuracy: 0.371	Validation Loss: 1.786 Validation Accuracy: 0.869	Validation Loss: 1.909 Validation Accuracy: 0.247	Validation Loss: 2.473 Validation Accuracy: 0.495
Fold 2	Validation Loss: 2.515 Validation Accuracy: 0.367	Validation Loss: 1.029 Validation Accuracy: 0.788	Validation Loss: 1.912 Validation Accuracy: 0.577	Validation Loss: 1.818 Validation Accuracy: 0.577
Fold 3	Validation Loss: 6.924 Validation Accuracy: 0.577	Validation Loss: 3.699 Validation Accuracy: 0.333	Validation Loss: 5.214 Validation Accuracy: 0.576	Validation Loss: 5.279 Validation Accuracy: 0.495
Fold 4	Validation Loss: 2.345 Validation Accuracy: 0.465	Validation Loss: 5.952 Validation Accuracy: 0.044	Validation Loss: 8.058 Validation Accuracy: 0.773	Validation Loss: 5.452 Validation Accuracy: 0.427
Fold 5	Validation Loss: 7.400 Validation Accuracy: 0.119	Validation Loss: 0.374 Validation Accuracy: 0.175	Validation Loss: 1.404 Validation Accuracy: 0.627	Validation Loss: 3.059 Validation Accuracy: 0.307
Fold 6	Validation Loss: 2.390 Validation Accuracy: 0.029	Validation Loss: 1.472 Validation Accuracy: 0.355	Validation Loss: 3.798 Validation Accuracy: 0.173	Validation Loss: 2.553 Validation Accuracy: 0.186

Table 5 BP4D-Spontaneous Dataset Female 6-Fold Testing

	Build 1	Build 2	Build 3	Average
Fold 1	Validation Loss: 0.722 Validation Accuracy: 0.676	Validation Loss: 2.242 Validation Accuracy: 0.668	Validation Loss: 0.503 Validation Accuracy: 0.391	Validation Loss: 1.155 Validation Accuracy: 0.579
Fold 2	Validation Loss: 3.330 Validation Accuracy: 0.623	Validation Loss: 3.143 Validation Accuracy: 0.403	Validation Loss: 2.727 Validation Accuracy: 0.050	Validation Loss: 3.067 Validation Accuracy: 0.359
Fold 3	Validation Loss: 3.351 Validation Accuracy: 0.284	Validation Loss: 2.399 Validation Accuracy: 0.233	Validation Loss: 1.454 Validation Accuracy: 0.331	Validation Loss: 2.401 Validation Accuracy: 0.283
Fold 4	Validation Loss: 7.806 Validation Accuracy: 0.445	Validation Loss: 2.656 Validation Accuracy: 0.738	Validation Loss: 3.914 Validation Accuracy: 0.223	Validation Loss: 4.792 Validation Accuracy: 0.469
Fold 5	Validation Loss: 3.006 Validation Accuracy: 0.615	Validation Loss: 1.053 Validation Accuracy: 0.147	Validation Loss: 3.035 Validation Accuracy: 0.599	Validation Loss: 2.364 Validation Accuracy: 0.454
Fold 6	Validation Loss: 0.545 Validation Accuracy: 0.721	Validation Loss: 0.609 Validation Accuracy: 0.293	Validation Loss: 2.059 Validation Accuracy: 0.047	Validation Loss: 1.071 Validation Accuracy: 0.354

Table 6 BP4D-Spontaneous Dataset Distribution Testing

	Build 1	Build 2	Build 3	Average
60 Male / 40 Female	Validation Loss: 2.049 Validation Accuracy: 0.668	Validation Loss: 0.833 Validation Accuracy: 0.714	Validation Loss: 1.632 Validation Accuracy: 0.625	Validation Loss: 1.504 Validation Accuracy: 0.669
40 Male / 60 Female	Validation Loss: 1.739 Validation Accuracy: 0.472	Validation Loss: 3.158 Validation Accuracy: 0.387	Validation Loss: 1.681 Validation Accuracy: 0.312	Validation Loss: 2.193 Validation Accuracy: 0.391
60 Pain / 40 No Pain	Validation Loss: 1.159 Validation Accuracy: 0.392	Validation Loss: 2.092 Validation Accuracy: 0.587	Validation Loss: 2.092 Validation Accuracy: 0.313	Validation Loss: 1.781 Validation Accuracy: 0.431
40 Pain / 60 No Pain	Validation Loss: 2.512 Validation Accuracy: 0.594	Validation Loss: 1.921 Validation Accuracy: 0.450	Validation Loss: 0.876 Validation Accuracy: 0.406	Validation Loss: 1.770 Validation Accuracy: 0.483

Chapter 5: Real-Time Pain Recognition on Hardware

The second portion of this study shifted focus to the hardware implementation of real-time pain recognition. The motivation was to produce a low-cost, compact deliverable which could reliably perform useful inferences as output to the user. Such a device has potential applications in a Telehealth setting or wherever the affect of participants is of particular interest.

5.1 Unit Costs and Specifications

The deliverable for this study consists of NVDIA's Jetson Nano single-board computer and a Raspberry Pi Camera Module V2. The Jetson Nano costs $99 USD and comes equipped with a 128-core NVIDIA Maxwell architecture-based GPU, a quad-core ARM A57 CPU, and a 4 GB 64-bit LPDDR4 memory. The size of the entire board is 100mm by 80mm [14]. Two power consumption modes are supported by the Jetson Nano, the MAXN mode which has a power budget of ten watts and the 5W mode which has a power budget of five watts [24]. The presence of a GPU on the board made the Jetson Nano a particularly attractive option for the study. A GPU, or graphical processing unit, performs parallel calculations on multiple sets of data. This approach to computations makes the hardware component beneficial to machine learning applications and IoT (Internet of Things) features. This is compared to the CPU, or central processing unit, which focuses on the throughput of calculations and is better equipped to execute parallel processes [4].

The Raspberry Pi Camera Module V2 costs $25 USD, weighs three grams, has a size approximately 25mm by 24 mm by 9 mm, and has an eight megapixels resolution. At most, the camera can capture up to 120 fps [2].

5.2 Model Implementation on Hardware

To deploy the pain recognition model on the Jetson Nano, the original Keras model was converted to an ONNX model format. ONNX, which stands for Open Neural Network Exchange, is an open format that allows for interoperability between various machine learning frameworks such as Caffe, Matlab, and PyTorch [16].

5.3 Hardware Performance

Assembled as a complete unit, the Jetson Nano board equipped with the Raspberry Pi camera and pain recognition model was able to make real-time inferences and provide feedback on the emotional state of the user. See figure 3 for real-time pain recognition inferences. With the Raspberry Pi camera module operating at 120 fps, the model made inferences using the live input from the camera with an elapsed time of 0.0121 s. The power consumption mode for the board set was set to MAXN. Setting the power consumption mode to 5W caused the board to experience undervoltage which led to significant performance degradation.

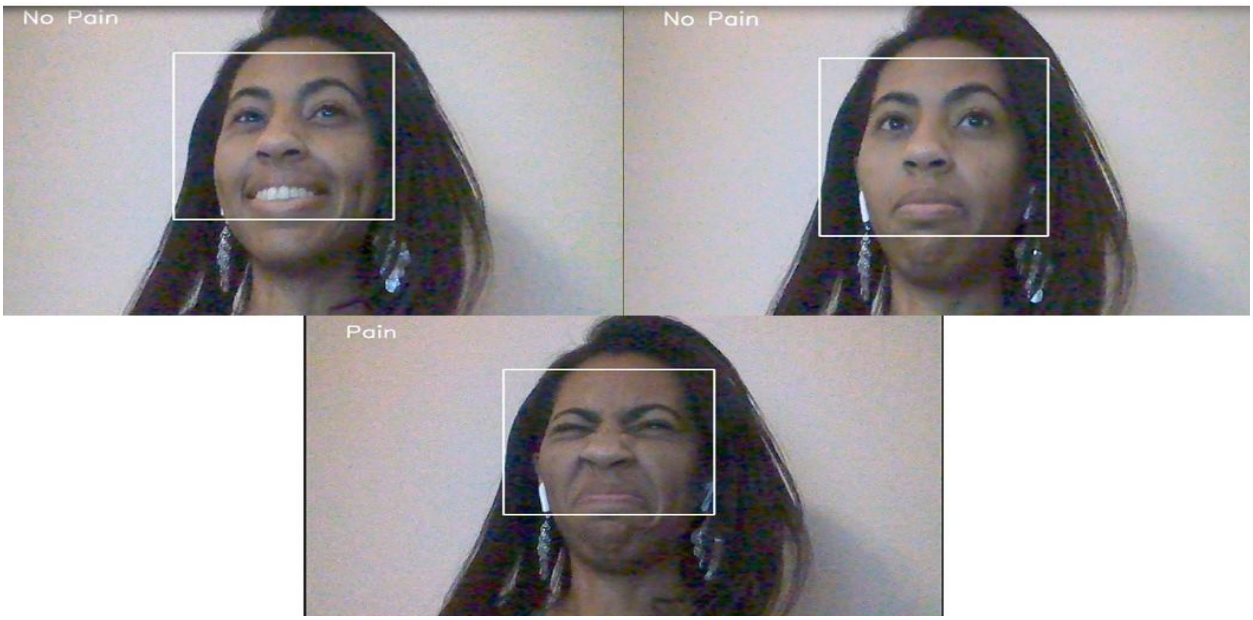

Figure 3 Real-time pain recognition inferences. Included with the owner's permission; which is the author of this paper.

Chapter 6: Conclusions and Future Work

6.1 Hardware Implementation

Real-time pain recognition is possible with the current state of technology and hardware. A unit capable of machine learning inference and image input collection was assembled from readily available components for less than $125 USD. See Figure 4 for hardware components shown individually and after assembled. The entire system has a power consumption comparable to one LED lightbulb, a tablet computer, or a Wi-Fi router [18].

Figure 4 Hardware components shown individually and assembled. Raspberry Pi Camera Module V2 (left), NVIDIA Jetson Nano (middle), unit assembled (right).

6.2 Pain-Recognition Modeling

Machine learning and modeling is the greatest hurdle to real-time pain recognition. The best model from the experimentation in this study produced a validation accuracy of seventy-two percent but had a validation loss value of 0.54. On average, the experiments which produced the highest performing models was the first experiment which conducted a 9-fold cross-validation

study and the fifth experiment in which one model was trained on a dataset that was sixty percent male and forty percent female. These findings suggest a dataset that is either balanced regarding gender or slightly male dominant is ideal for pain-recognition model training. These results also show while some imbalance involving gender is acceptable, the distribution of 'pain' and 'no pain' classes should remain balanced for the best performance. In all experiments, however, the validation loss remained high which signals achieving generalized learning, rather than memorization, is still a challenge.

Implicit bias when creating the BP4D dataset subset could have also had an impact on the performance of the pain recognition modeling. The BP4D dataset subset was curated by one individual: the author of this study. Humans unconsciously, and at times consciously, are influenced by their culture, age, and personal experiences in determining what is the most "appropriate" or "likely" physical outward expression of an emotion. This issue of built-in bias highlights a larger challenge to automated pain-recognition across cultures, demographics, and individual personalities. This is a challenging and prevalent problem in affective computing and the larger artificial intelligence field [21]. This is out of scope of this thesis and left for future work.

6.3 Future Work

Classification based on multimodal input data appears to be the best path forward for the study of pain recognition [32]. In addition to visual data, research has been done to incorporate audio data, as well as physiological data such as EEG signals. To handle this increased number of diverse inputs, additional peripherals would need to be added to a deliverable, for example the Jetson Nano used in this study. Expanding the total amount of modules needed to make a pain recognition system work could have a negative impact on power consumption, assembly, and size, creating a deliverable which is more expensive and complicated to construct and ultimately less

accessible to individuals. Considering this, future work will investigate the cost versus power consumption of such as device, while considering the overall boost in accuracy.

References

[1] Cambria, Erik, Dipankar Das, Sivaji Bandyopadhyay, and Antonio Feraco. "Affective computing and sentiment analysis." In A practical guide to sentiment analysis, pp. 1-10. Springer, Cham, 2017.

[2] “Camera Module,” Camera Module - Raspberry Pi Documentation, 2016. [Online]. Available: https://www.raspberrypi.org/documentation/hardware/camera/. [Accessed: 06-Mar-2021].

[3] Cootes, T. F., Edwards, G. J., & Taylor, C. J. “Active appearance models. In European conference on computer vision, 1998.

[4] “CPU vs GPU: Definition and FAQs,” OmniSci. [Online]. Available: https://www.omnisci.com/technical-glossary/cpu-vs-gpu. [Accessed: 07-Mar-2021].

[5] D. C. D. (DCD), “Opioid Crisis Statistics,” HHS.gov, 2019. [Online]. Available: https://www.hhs.gov/opioids/about-the-epidemic/opioid-crisis-statistics/index.html#:~:text=In%202019%2C%20an%20estimated%2010.1,and%20745%2C000%20people%20used%20heroin.&text=Appropriate%20prescribing%20of%20opioids%20is,and%20safety%20of%20Medicare%20beneficiaries. [Accessed: 02-Mar-2021].

[6] E. A. Taub, “Sleepy Behind the Wheel? Some Cars Can Tell,” The New York Times, 16-Mar-2017. [Online]. Available: https://www.nytimes.com/2017/03/16/automobiles/wheels/drowsy-driving-technology.html. [Accessed: 01-Mar-2021].

[7] Ekman, P., & Friesen, W. V. “Facial action coding system: Investigator's guide,” Consulting Psychologists Press, 1978.

[8] Fabiano, D., & Canavan, S. “Emotion recognition using fused physiological signals,” International Conference on Affective Computing and Intelligent Interaction, 2019.

[9] Fleureau, J., Guillotel, P. and Huynh-Thu, Q., “Physiological-based affect event detector for entertainment video applications,” IEEE Transactions on Affective Computing, 3(3), pp. 379-385, 2012

[10] Hinduja, S., Uddin, M. T., Jannat, S. R., Sharma, A., & Canavan, S. “Fusion of hand-crafted and deep features for empathy prediction,” IEEE International Conference on Automatic Face & Gesture Recognition, 2019.

[11] Jannat R. Sk, Tynes, I., Lime, L., Adorno, J., and S. Canavan. “Ubiquitous emotion recognition using audio and video data.” International Joint Conference and International Symposium on Pervasive and Ubiquitous Computing and Wearable Computers, 2018.

[12] K. Z. JG. Guy, D. H. A. Schuchat, J. N. K. JM. Donohue, A. K. M. Burden, M. B. R. SJ. Herzig, D. W. Baker, D. R. A. Zgierska, A. D. D. SL. Calcaterra, P. S. A. Tong, N. T. J. Green, E. M.-C. J. Fereday, L. A. C. EH. Bradley, W. L. M. BF. Crabtree, J. S. B. Saunders, B. C. J. Borkan, K. Charmaz, J. D. D. Davies, C. T. W. K. Hinami, K. D. PI. Buerhaus, M. A. R. A. Wagner, S. L. S. Singer, K. K. JL. Jackson, M. J. G. CM. Döpp, K. J. W. LE. Sinnenberg, S. B. AG. Bot, S. A. S. SP. Nota, W. W. DP. Maher, T. E. Y. SL. Calcaterra, S. R. H. JH. Donroe, D. B. G. R. Chou, J. N. H. M. Yajnik, A. F. M. F. Balagué, D. J. Burgess, S. G. M. Achkar, A. M. G. DM. Dave, and J. M. H. H. Wen, "Clinical perspectives on hospitals' role in the opioid epidemic," BMC Health Services Research, 01-Jan-1970. [Online]. Available: https://bmchealthservres.biomedcentral.com/articles/10.1186/s12913-020-05390-4#citeas. [Accessed: 02-Mar-2021].

[13] Lucey, P., Cohn, J., Lucey, S., Matthews, I., Sridharan, S., & Prkachin, K. M. "Automatically detecting pain using facial actions," International Conference on Affective Computing and Intelligent Interaction and Workshops, 2009

[14] N. V. I. D. I. A. Newsroom, "NVIDIA Announces Jetson Nano: $99 Tiny, Yet Mighty NVIDIA CUDA-X AI Computer That Runs All AI Models," NVIDIA Newsroom Newsroom, 24-Feb-2021. [Online]. Available: https://nvidianews.nvidia.com/news/nvidia-announces-jetson-nano-99-tiny-yet-mighty-nvidia-cuda-x-ai-computer-that-runs-all-ai-models. [Accessed: 06-Mar-2021].

[15] O. D. George W. Meers, "Learn to Spot Drug-Seeking Behavior," Review of Optometry, 15-Feb-2014. [Online]. Available: https://www.reviewofoptometry.com/article/learn-to-spot-drugseeking-behavior. [Accessed: 02-Mar-2021].

[16] “Open Neural Network Exchange,” ONNX. [Online]. Available: https://onnx.ai/. [Accessed: 06-Mar-2021].

[17] Picard, R. “Affective Computing,” MIT press, 2000.

[18] “Power Consumption of Typical Household Appliances,” Daft Logic, 2020. [Online]. Available: https://www.daftlogic.com/information-appliance-power-consumption.htm. [Accessed: 07-Mar-2021].

[19] R. Gelburd, “Telehealth Claim Lines Increase in October 2020 as COVID-19 Pandemic Surges,” AJMC, 07-Jan-2021. [Online]. Available: https://www.ajmc.com/view/telehealth-claim-lines-increase-in-october-2020-as-covid-19-pandemic-surges. [Accessed: 02-Mar-2021].

[20] Roeder, Lutz. Netron, 2020, netron.app/.

[21] Roselli, D., Matthews, J., & Talagala, N. “Managing bias in AI,” Companion Proceedings of the 2019 World Wide Web Conference, 2019.

[22] S. Albawi, T. A. Mohammed and S. Al-Zawi, "Understanding of a convolutional neural network," 2017 International Conference on Engineering and Technology (ICET), Antalya, Turkey, 2017, pp. 1-6, doi: 10.1109/ICEngTechnol.2017.8308186.

[23] S. Bargal , E. Barsoum, C. Ferrer, and C. Zhang, “Emotion recognition in the wild from videos using images,” ICMI '16: Proceedings of the 18th ACM International Conference on Multimodal Interaction, pp. 433–436, Oct. 2016.

[24] Tegra Linux Driver, 02-Feb-2021. [Online]. Available: https://docs.nvidia.com/jetson/l4t/index.html#page/Tegra%20Linux%20Driver%20Package%20Development%20Guide/power_management_nano.html. [Accessed: 06-Mar-2021].

[25] U.S. Department of Health and Human Services, “HHS Issues New Report Highlighting Dramatic Trends in Medicare Beneficiary Telehealth Utilization amid COVID-19,” HHS.gov, 21-Jan-2021. [Online]. Available: https://www.hhs.gov/about/news/2020/07/28/hhs-issues-new-report-highlighting-dramatic-trends-in-medicare-beneficiary-telehealth-utilization-amid-covid-19.html. [Accessed: 02-Mar-2021].

[26] Uddin, M. T., Canavan, S., & Zamzmi, G. “Accounting for Affect in Pain Level Recognition,” arXiv preprint arXiv:2011.07421, 2020.

[27] “Understanding the Epidemic,” Centers for Disease Control and Prevention, 19-Mar-2020. [Online]. Available: https://www.cdc.gov/drugoverdose/epidemic/index.html. [Accessed: 02-Mar-2021].

[28] Werner, P., Al-Hamadi, A., Niese, R., Walter, S., Gruss, S., & Traue, H. C. “Automatic pain recognition from video and biomedical signals,” 22nd International Conference on Pattern Recognition, pp. 4582-4587), 2014.

[29] Xing Zhang, Lijun Yin, Jeff Cohn, Shaun Canavan, Michael Reale, Andy Horowitz, Peng Liu, and Jeff Girard, “BP4D-Spontaneous: A high resolution spontaneous 3D dynamic facial expression database”, Image and Vision Computing, 32 (2014), pp. 692-706 (special issue of the Best of FG13)

[30] Xing Zhang, Lijun Yin, Jeff Cohn, Shaun Canavan, Michael Reale, Andy Horowitz, and Peng Liu, “A high resolution spontaneous 3D dynamic facial expression database”, The 10th IEEE International Conference on Automatic Face and Gesture Recognition (FG13), April, 2013.

[31] X.-W. Wang, D. Nie, and B.-L. Lu, “Emotional state classification from EEG data using machine learning approach,” Neurocomputing, vol. 129, pp. 94–106, Apr. 2014.

[32] Zamzmi, G., Kasturi, R., Goldgof, D., Zhi, R., Ashmeade, T., & Sun, Y. “A review of automated pain assessment in infants: features, classification tasks, and databases,” IEEE reviews in biomedical engineering, 2017.

[33] Zhang, Z., Girard, J.M., Wu, Y., Zhang, X., Liu, P., Ciftci, U., Canavan, S., Reale, M., Horowitz, A., Yang, H. and Cohn, J.F. “Multimodal spontaneous emotion corpus for human behavior analysis,” IEEE Conference on Computer Vision and Pattern Recognition, 2016

[34] Zheng W., Tang H., Lin Z., Huang T.S. (2010) Emotion Recognition from Arbitrary View Facial Images. In: Daniilidis K., Maragos P., Paragios N. (eds) Computer Vision – ECCV 2010. ECCV 2010. Lecture Notes in Computer Science, vol 6316. Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-642-15567-3_36