МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ МОСКОВСКИЙ ФИЗИКО-ТЕХНИЧЕСКИЙ ИНСТИТУТ (ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ)

ФАКУЛЬТЕТ УПРАВЛЕНИЯ И ПРИКЛАДНОЙ МАТЕМАТИКИ
ВЫЧИСЛИТЕЛЬНЫЙ ЦЕНТР ИМ. А. А. ДОРОДНИЦЫНА РАН
КАФЕДРА «ИНТЕЛЛЕКТУАЛЬНЫЕ СИСТЕМЫ»

Кудряшова Александра Дмитриевна

Определение интенсивности эмоций по видео

010656 — Математические и информационные технологии

ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА МАГИСТРА

Научный руководитель:

д. ф.-м. н. Стрижов Вадим Викторович

Москва

2015

Содержание

Аннотация

В работе описана система по удаленному измерению пульса и интенсивности и эмоций, а также исследована корреляция между их значениями. Проект состоит из трех частей. В первой разрабатывается интерактивная система для получения данных от веб-камеры и измерение пульса как частоты изменения интенсивности зеленого канала изображения. Вторая - серия классификаторов, распознающих интенсивность эмоций. В третьей части построена корреляция значений пульса и интенсивности эмоций.

Проект основан на предположении, что существует корреляция между изменением сердечного ритма и интенсивностью эмоций. Для проверки этого предположения было предложено построить систему, где пользователю показывается видео-стимул, в это время его лицо записывается с помощью веб-камеры и проводится анализ этой записи. Результатом проекта является онлайн-система, распознающая пульс, интенсивность одной из четырех эмоций: счастье, гнев, удивление, грусть, показывающая пользователю видеозапись и одновременно записывающая лицо пользователя, а также насчитывающая частоту сердечных сокращений. Система была протестирована на десяти респондентах.

Введение

Предметная область и постановка задачи

Объем использования Интернетом растет из года в год, и стало возможным реализовать такой новый сегмент технологий как получение удаленных измерений физического состояния человека. В моей работе, я создала систему дистанционного определения интенсивности эмоций с помощью измерения физического состояния, полученного из видеозаписи. Проблему определения пульса также решают некоторые компании и научно-исследовательские центры. Количество проектов, в которых специалисты удаленно решать различные проблемы и предоставить консультации возросло с развитием телекоммуникаций. В частности, врачи различных специальностей могут использовать видеоконференции или видео-обследование пациентов для постановки или уточнения диагноза, контроля за ходом заболевания и результатами лечения.

Помимо очевидного использования видео для обследования пациента, есть способы использования записи серии изображений или видео для более сложных измерений. Примером таких измерений могут быть измерение частоты сердечных сокращений, частоты дыхания и изменений размера зрачка. Особый интерес представляет получение изображений, которое не только позволяет определить физическое состояние человека в данный момент, но и судить о более высокоуровеных параметрах. Примером таких характеристик может быть настроение человека, его физическое состояние, возбужденность или усталость.

Кроме того, измерение импульса с помощью серии изображений или видео потока необходимо в таких областях, как маркетинг. Маркетологи могут получить значение физических измерений для определения эффективности рекламы, развлечений и виртуальных видеоигр, чтобы определить настроение пользователя и предложить оптимальный сценарий. Необходимо также упомянуть систему видеонаблюдения для машинистов и водителей коммерческого транспорта - для них также разработаны системы наблюдения для предотвращения потери контроля, потому что водитель заснул.

В моей работе я буду решать проблему дистанционного измерения физических параметров таких, как частота сердечных сокращений и разработаю способ прогнозирования эмоции человеческие, используя эти данные. Проблема состоит из нескольких частей:

- разработки онлайн инструмента для записи данных от веб-камеры,
- построения алгоритма для получения синхронных измерений серии изображений,
- получения показаний пульса из видео
- алгоритм машинного обучения для определения типа и интенсивности эмоций пользователя.

Цель и термины

Цель проекта - проверить возможность разработки системы определения и прогнозирования эмоций на основе видео, полученного с веб-камеры путем измерения показателей физического состояния человека. В работе будут использованы следующие термины:

Физическое состояние - быстро изменяемые параметры тела человека, например ЧСС, частота дыхания, размер зрачка глаза;

эмоции - субъективный, сознательный опыт характеризующийся психофизиологическими реакциями, биологическими реакциями, и психическим состоянием;

прогнозирование эмоций - классификация эмоций пользователя в один из двух классов: нейтральный и удивление;

видео, полученные с веб-камеры - короткие (15-30 секунд) отрывки видеоролика, полученные с помощью веб-камеры для анализа физического состояния;

Информационная система - это система, объединяющая людей и компьютеры с целью получения, обработки или интерпретации информации.

синхронизированные измерения - измерения, полученные от информационной системы, спроектированной таким образом, что несколько процессов различной природы обрабатываются определенным образом, при этом все изменения физического состояния человека можно легко и точно отследить.

Обзор литературы

Чтобы подойти к решению поставленной проблемы я рассмотрела 20 источников, обсуждающих теорию и практику получения и усиления сигналов для обнаружения сердечного ритма, методы и приемы формализации, обнаружения и интерпретации эмоций, а также подготовила обзор современных систем распознавания эмоций. Некоторые ресурсы представляют собой научные работы, опубликованные на конференции AVEC (Audio-Visual Emotion recognition Challenge) - крупнейшем событии в области систем распознавания эмоций. Я добавила описания наиболее актуальных для моего проекта работ в этой области.

Перечисленные документы и проекты решают ряд актуальных задач в области анализа выражения лица и распознавания эмоций. Первый проект заключается в разработке унифицированного способа кодирования эмоций и мимики. Один из основных подходов к этой проблеме описан в Multimodal Emotion Recognition in Response to Videos [1]. Эта статья представляет собой объективный метод распознавания эмоций пользователя, с целью восстановления тегов в видео с помощью электроэнцефалограммы (ЭЭГ) и показаний реакции от сетчатки пользователя. Для выполнения работы, были выбраны 20 видео клипов, содержащих эмоционально-насыщенные фильмы и интернет-ролики. Для этих клипов были записаны показания ЭЭГ и изменения размера зрачка у 24 участников во время просмотра фильмов. Исходные значения были определены на основе медианного значения возбуждения и валютных баллов, присвоенных клипам в предварительном исследовании, с использованием онлайн-анкет. На основании ответов участников, были определены три класса для каждого измерения. Эти классы заставить интенсивности: ”нейтрально", "среднее волнение" и “эмоциональный подъем" и валентные классов - “плохо", “хорошо”, “средне”.

При оценке влияния различных мультимедийных материалов на эмоции при обучении эффективность визуальных и вербальных способов подачи материала обсуждается в Assessing the effects of different multimedia materials [2]. Мультимедийные материалы в настоящее время все чаще используется в образовательных программах. Тем не менее, индивидуальные предпочтения мультимедийных материалов на основе визуальных и вербальных когнитивных особенностей могут влиять на эмоции и результаты учащихся. Таким образом,в работе исследовано как различные мультимедийные материалы влияют на эффективность обучения и эмоции учащихся. Кроме того, многие ученые утверждают, что эмоции оказывают непосредственное влияние на эффективность образования. Таким образом, дальнейшие исследования, которые подтверждают взаимосвязь между эмоциями и успеваемостью учащихся дают прикладные знания для проектирования адаптивной мультимедиа системы обучения с индивидуальным подходом.

Компанией HeartMath был разработан инструмент эмоций emWave, который, используется для оценки различных эмоциональных состояний для вербальных и визуальных эффектов в процессе обучения. Три различные способа представления массовой информации: статический текст, изображения на основе мультимедийных видео материалов, интерактивных и анимированные мультимедийные материалов были рассмотрены. Целью ставилось исследовать, как влияют различные способы представления медиа-материалы на производительность индивидуального обучения и на настроение обучающихся, а также определить взаимосвязь между обучением и эмоциями. Экспериментальные результаты показывают, что мультимедийные видео материалы предлагают лучшую производительность. Кроме того, динамический контент, содержащих видео и анимацию больше подходят для визуального чем для статического медиа-контента, содержащего текст и изображения. Наконец, есть частичная корреляция между негативными эмоциями и обучением.

В Using emotion recognition technology to assess the effects of different multimedia materials on learning emotion and performance [3] авторы поднимают вопрос о признании интенсивности эмоций в качестве параметра для сравнения различных источников обучения. С постепенным введением мультимедийных технологий в образовательный процесс, увеличилась потребность в углубленном исследований того, как разные способы предоставления мультимедийных материалов влияют на эмоции и успеваемость учащихся. Это исследование привело к разработке системы emWave - детектора интенсивности эмоциональных состояний.

На предварительной стадии, исследователи проанализировали собранные данные и эмоциональную оценку производительности обучения. В исследовании они рассмотрели, как различные мультимедийные учебные материалы влияют на эмоции, и в конечном итоге на успеваемость. Предварительные результаты показали, что мультимедийный учебный материал на основе видео обеспечивает наилучшую эффективность и самые положительные эмоции среди трех типов мультимедийных материалов оцененных в данном исследовании. Кроме того, частичная корреляция между негативными эмоциями и производительностью была отмечена. Это исследование подтверждает, что эмоции могут предсказать эффективность обучения студентов, если используется видео на основе мультимедийных материалов для изучения.

В FaceFetch: A User Emotion Driven Multimedia Content Recommendation System Based on Facial Expression Recognition [4] приведен пример такой системы. Распознавание мимики пользователей позволяет исследователям строить контекстно-зависимых приложения, которые адаптируются в соответствии с эмоциональным состоянием пользователя. Распознавание мимики является активной областью исследований в сообществе компьютерного зрения. В этой статье про FaceFetch новая система контекстных рекомендаций мультимедийного контента распознает текущее эмоциональное состояние пользователя (счастье, печаль, страх, отвращение, удивление и гнев) через декодирование мимики. Эта система может понимать эмоциональное состояние пользователя через настольного компьютера, а также с помощью мобильного пользовательского интерфейса и выбрать медиа-контента, такие как музыка, фильмы с учетом интересов пользователя в режиме реального времени.

В Emotion detection using sub-image based features through human facial expressions [5], авторы рассматривают методы кодирования и предлагают общий подход к распознаванию лица выражений. Лицо человека - важная часть человеческого тела, которая играет решающую роль во взаимодействии людей и человека с компьютером. Поэтому важно разработать надежную систему для обнаружения эмоций для применение в широком спектре задач, например, для эффективного взаимодействия человека и компьютера. Выражение лица обеспечивает невербальное общение. В исследовании выявляется проблема потери данных при признаков по причине ограниченного числа закодированных позиций лицевых мышц. Для улучшения производительности обнаружения предлагается рассматривать относительную производительность. Классификация осуществляется с помощью автоматизированной системы обнаружения эмоций и мимики, использующей классификатор на основании SVM. Результаты показывают, что предложенная относительная производительность повышает качество классификации

В Dynamics of Facial Expression Extracted Automatically from Video [6], авторы обсуждают дизайн системы распознавания эмоций, принимающей видеопоток в качестве входных. Работа представляет собой систематическое сравнение методов машинного обучения, применяемые к проблеме полностью автоматического распознавания мимики, в том числе AdaBoost, SVM, и LDA. Каждый видеокадр предварительно записывается в режиме реального времени для обнаружения фронтальной проекции лица в вертикальном положении. Лица масштабируется до изображений фиксированного размера, сжатых фильтром Габора, а затем кодируются, по семи измерениям в режиме реального времени: нейтральное, гнев, отвращение, страх, радость, печаль, удивление. В работе сообщается о результатах серии экспериментов. Наилучшие результаты были получены путем выбора подмножества фильтров Габора, используя AdaBoost, а затем тренировка SVM на выходах фильтра, выбранного AdaBoost в качестве признаков. Точность алгоритма для синтетических данных составила 93%, а для открытых баз данных стала еще выше. Неожиданно было обнаружено, что в регистрации структурных особенностей лица не было необходимости, поскольку узнавание человека не обеспечивают точность обнаружения для эмоций.

В Fully Automatic Facial Action Recognition in Spontaneous Behavior [7] та же группа авторов делает дополнительную работу, чтобы предложить способ интерпретировать мимику человека единым образом. Результаты разработки полностью автоматического распознавание лица в режиме реального времени представлены в виде система кодирования FACS, которая может стать отраслевым стандартом в области кодирования выражений лица. Система автоматически определяет фронтальную проекцию лица в видео-потоке и кодирует каждый кадр с помощью 20 единиц действия (action units). Предварительные результаты решения проблемы обнаружения мимики в спонтанных выражениях во время разговора. сравнивается после обработки классификаторами SVM и AdaBoost. Наиболее интересные находки в этой работе объясняется тем, что наилучшие результаты в распознавании эмоций были получены в 3 шага машинного обучения, где AdaBoost используется на первом этапе, фильтрация Габора на второй и SVM делается на выходах фильтра Габора.

Some practice implementation applicable to the real life problem was discussed in Recent developments in openSMILE [8]. openSMILE - популярная библиотека, которая извлекает признаки из мультимедийного сигнала лица человека. Версия 2.0 объединяет объект распознавания речи, музыку, звук и общие события с основными функциями видео для мультимодальных средств. Аудио и видео дескрипторы (описания) могут быть обработаны все вмесие, онлайн обработки и оффлайн пакетной обработкой, и в результате могут быть получены статистические функциональны, такие как точки, пики и т.д. Они включают в себя статистические классификаторы, такие как SVM и могут экспортировать файл для дальнейшей обработки. Доступные дескрипторы низкого уровня включают в себя речь, музыку и видео функции. Системы быстро работает на Unix и Windows платформах.

To move forward with the development part, the ideas and methods from Error Weighted Semi-Coupled Hidden Markov Model for Audio-Visual Emotion Recognition [9] were reviewed. This paper presents an approach to automatic recognition of human emotions through the audio-visual signals using a bimodal EWSC-HMM. The proposed approach combines the SC-HMM-based states bimodal alignment and Bayesian classifier to obtain optimal results emotion recognition based on audio-visual bi-modal synthesis. A bimodal strategy based on states, alignment SC-HMM, is designed to harmonize the temporary connection between the audio and visual streams. Bayesian classifier is then used to investigate the contribution of SC-HMM-based classifiers for a variety of audio-visual couples to get certain emotions. To assess the quality of the work, the two databases are read: MHMC database and Semaine database. Experimental results show that the proposed approach is not only superior to other methods of emotion recognition.

The most important event in the world of emotion recognition is AVEC. The results of AVEC 2011, 2012 and 2013 are discussed below.

In AVEC 2013: the continuous audio/visual emotion and depression recognition challenge [10] notes the methods for online emotion recognition of video. The worsening of mood associated with emotions. In particular, the behavior of people who suffer from mood disorders such as unipolar depression shows a strong temporal correlation with the size of affective valence and arousal. Also, psychologists and psychiatrists watching facial expressions and tone of voice in the assessment of the patient's condition. Depression can lead to expressive behavior, such as slurred facial expressions, avoiding eye contact and the use of short sentences with a flat tone. The AVEC 2013 had two objectives: first, to predict continuous values affective dimensions valence and arousal at each time point. Second one predicts the value of a single indicator of depression for each record in the dataset. The result of the work was in the system that predicts depression based on the formalized surveys and Local Phase Quantisation (LPQ) as that was found to attain higher performance in facial expression recognition tasks.

In a Multimodal Fuzzy Inference System Using a Continuous Facial Expression Representation for Emotion Detection [11], the results of AVEC 2012 are presented. This paper presents a multi-modal system of fuzzy inference to detect emotions. The system retrieves and combines the visual, auditory and contextual parameters. Experiments were performed under AVEC 2012. Mimicry plays an important role in detecting emotions. And having an automated system for the detection of facial expressions from unknown actors is a challenge. A method that adapts to the morphology of the person and which is based on invariant representations of facial expressions. The method is based on the eight key expressions of emotion. In the system, each video image sequence is determined by the relative expression of these 8. 8 These expressions are synthesized for each subject through a neutral person subject. Emotion is described in the four dimensions: valence, arousal, intensity, and duration. The results show that the duration of high intensity is smiling expression that makes sense for continuously detecting valences and can also be used to improve the detection of excitation. Major changes in the power and duration are given according to the context.

Meta-Analysis of the Facial Expression Recognition Challenge [12]. Automatic recognition of expression was an active topic in computer science for two decades, in particular, the coding system of facial expressions, as well as the definition and classification of some discrete emotions. Standardization and comparability received some attention; For example, there are some widely used database of facial expressions. Nevertheless, the lack of a common protocol and evaluation, the lack of sufficient details that are necessary to reproduce lead to the fact that it is difficult to compare the results of individual independent. This, in turn, hinders progress in this field. AVEC initiative in 2011 attempted to solve the problem of standardization and described the typical elements of facial expressions. At the moment, they are the de facto standard in the field. In this project, a subset of FACS system of facial expressions coding was used including only “Surprise” and “Neutral” classes was found. This work is a great source of feature extraction ideas.

The following findings allow to get deeper in methods and techniques of filtering and machine learning algorithms such as AdaBoost, Gabor Filter, SVM, Magnification.

In Generalized Multiclass AdaBoost and Its Applications to Multimedia Classification [13] presented the features and applications of AdaBoost algorithm used in the project. AdaBoost has received considerable attention from the media, the research community in recent years. It was originally designed for classification problems with two classes. To cope with the number of classes has been developed many extensions of AdaBoost From a statistical point of view, AdaBoost can be seen as a direct step of the additive model using the exponential function loss. In this article, we obtained generalized form AdaBoost for multiclass classification based on the exponential function loss. To prove the effectiveness of, it has been tested some multimedia tasks of different nature. Experimental results show that the new algorithm is superior to other improve. Also, the generic algorithm increase can be used to build a multiclass classifier from binary.

AdaBoost achieves boosting by combining many “weak” classifiers (ht) to produce a “strong” classifier (H):

$$H(x) = \sum_t \alpha_t h_t(x)$$

For a K-class classification problem, we assume a set of training samples $\{(x_i, y_i), i = 1,...,m\}$ are given, where x_i belongs to a domain and y_i is the label of instance i. Let $U^k = \{u_1,...,u_K\}$ be the unit base vectors of a K-dimension space R^K. The labels can be represented by a base vector, i.e., $y_i = u_n$ if the instance i is in class n. The classifier h(x) produces a vector belonging to R^K, or $h(x) \in U^K$ if h(x) is discrete. Final algorithm is presented below:

```
1 Initialize the weight $w_{ij}$ ($i = 1,...,m$ and $j = 1,...,K$):
    $w_{ij} = \begin{cases} 0 & y_i = u_j \\ 1 & otherwise \end{cases}$
2 For $t = 1,...,T$:
    a. Normalize $w_{ij}$.
    b. Train $h_t(x)$ by minimizing loss function:
        $L = \sum_{i=1}^{m} \sum_{j=1}^{K} w_{ij} \exp((u_j - y_i) \cdot h_t(x_i))$    (1)
    c. Update the weight matrix $w_{ij}$:
        $w_{ij} \leftarrow w_{ij} \exp((u_j - y_i) \cdot h_t(x_i))$    (2)
3 Final classifier:
    $H(x) = \sum_{t=1}^{T} h_t(x)$
```

The parts of Theoretical Views of Boosting and Applications [14] were used for better understanding of the application of boosting in real life projects. Busting a general method for improving the accuracy of a learning algorithm. Focusing mainly on the algorithm AdaBoost, a brief review of theoretical works on boosting, including the analysis of errors in the AdaBoost learning and the generalized error, and boosting communication between game theory, methods of determining the probability of the use of extensions and boosting AdaBoost to solve multiclass classification. The most valuable part of the work is an approach for error estimation.

Training error:

AdaBoost is a procedure for finding a linear combination *f* of weak hypotheses which attempts to minimize

$$\sum_{i'} exp\left(-y_i f(x_i)\right) = \sum_{i} exp\left(y_i \sum_{t} \alpha_t h_t(x_i)\right)$$

Generalization error:

$$\widehat{Pr}\left[\frac{y\Sigma_t \alpha_t h_t(x)}{\sum_t |\alpha_t|} \leq \theta\right] + \bar{O}\left(\frac{\sqrt{d}}{m\theta^2}\right)$$

Real-Time Face Detection and Facial Expression Recognition: Development and Applications to Human-Computer Interaction [15] is a great example of a review of all the tasks that can be performed using Facial Expression Recognition. The most important of them is the assessment of emotions. Computer animated program and bring robots to the social aspect of human-computer interaction, and forced to think in new ways about how computers can be used in everyday life. Face-to-face real time is working for about 40 milliseconds. The level of uncertainty in the timeline considerably, making it necessary for man and machine, relying on sensory perception primitives, and not the slow process of symbolic output. The paper presents the progress in one of these projects through the perception of primitives. The system automatically detects the front face in the video stream and encodes them to 7 measurements in real time: neutral, anger, disgust, fear, joy, sadness, surprise. Face recognizer uses a cascade of filters, boosting trained. Image Recognizer receives samples of the image. Gabor representation is then processed by the bank SVM classifiers. The new combination and AdaBoost SVM performance improvement system. The system has been tested on Cohn-Kanade dataset related facial expressions. In this work, the researchers suggest a way how to identify a face for further analyzing by using and Integral image filter.

To decrease the dimensions of the tasks, researchers often use Gabor transformation. In Gabor wavelet transform and its application [16], they discuss how to prepare data for further transformation and how to assess the output of the filter. This report is dedicated to wavelet transform and Gabor filter applications. The wavelet transform can extract time (spatial) and the frequency data from the signal. Among the types of wavelet transforms, Gabor wavelet transform has impressive mathematical and biological properties and is often used in imaging studies.

The next step of the implementation of the system is visualizing the inputs and outputs of Gabor Filter. In Gabor Filter Visualization [17] paper, the tools and best practices of visualization are discussed. The biggest value of the work is in using the shape of the graph also to valued parameters. A system for imaging an important signal processing techniques - Gabor filter. To do this, you need to overcome the problem of representation of multi-dimensional space on a static graph. We used an interactive widget to change the visible area of the predicted sizes, as well as additional charts that summarize the answers in the projected dimension. Therefore, seeing this four-space through 2-dimension projection can understand all aspects, and not just the projection plane. It was found that the implemented system had helped to understand better the Gabor filter. On figure 8 an example of visualization system is presented. In that particular example, the system calculated the frequency of appearance of different colors and frequency of appearance of the lightest pixels. Lower right image is minimized and ready to be encoded.

The most popular static dataset of labeled emotions is Cohn-Kanade dataset [18]. Database coded facial expressions Cohn-Kanade used for research in the field of the automatic face, such as analysis and synthesis of image and perception surveys. Cohn-Kanade is available in two versions, and the third is in preparation.

Version 1, the first issue includes a sequence of 97,486 models. Each sequence starts with a neutral expression and proceeds to a peak of expression. Peak expression for each sequence encoded tag given emotion. Tag emotions means that the expression of emotions was proposed but not that it really can be done.

Version 2 (CK +), it includes both staged and spontaneous expression, and additional types of metadata. To create expressions, the number of sequences is increased by 22% and the number of subjects by 27%. As in the first version, the expression for each target sequence is completely marked. Also tested were added e metadata tag proven emotions. Thus, the sequence can be analyzed for and action blocks and prototypical emotions. Also, the IC + provides the protocols and underlying results for the face-tracking features, and the unit of action and emotion recognition. To determine the actions and expressions has been used linear SVM classifier (2010).

The most promising method of identifying pulse by video record is Eulerian Video Magnification. It is discussed in [19]. They aimed to detect temporal variations in videos that are difficult and even impossible to identify with the human's eye without additional devices and sensors. The method call Eulerian Video Magnification takes a standard video sequence (RGB) as input and applies the spatial decomposition to its channels. Temporal filtering is later applied against the outputs of spatial decomposition. The resulting meaningful signal is amplified to reveal hidden information after that.

This method allows to see blood as it fills the face and also to amplify and reveal small

motions. Blood vibration can be recalled as heart rate so that magnification allows to measure heart rate. As a result they completed four steps:

(1) selected a temporal bandpass filter;

(2) selected an amplification factor, α;

(3) selected a spatial frequency limit (specified by spatial wavelength, λc) beyond which an attenuated version of α is used;

(4) selected the shape of the attenuation for α—either force α to zero for all $\lambda < \lambda c$, or linearly scale α down to zero.

The wavelength band of concern can be chosen automatically in some states, but it is often important for users to check the frequency band corresponding to their purpose. In our real-time application, the amplification factor and cutoff frequencies are all configurable by the user.

They checked the performance of the method for color amplification using two videos of adults of different nations and one of a baby. An adult subject with lighter complexion is shown in *face1* (Figure 10) while an individual with darker complexion is shown in *face2*. For both samples, the objective was to increase the color change as the blood passes through the face. In *face1* and *face2*, a Laplacian pyramid and set α for the finest two levels to 0 were used. They downsampled and filtered with a spatial lowpass each block to reduce noise and to increase the implied pulse signal.

For each video, they passed each sequence of frames through a bandpass filter with a band of 0.83 Hz to 1 Hz. Finally, they applied a large value of $\alpha \approx 100$ and $\lambda c \approx 1000$ to highlight the tone change. The final video was produced by adding the data back to the source. They observed periodic green to red movements at the heartbeat rate. *baby* is a video of a newborn recorded in situ. In addition to the video, they collected vital signs from a hospital grade monitor. They used this data to prove the truth of the heart rate estimation and to verify that the color amplification signal.

Pulse detection

Theory:

The human visual system has a limited space-time sensitive, but many signals which are out of range of perception may be informative.

For example, human skin color varies somewhat with the strengthening of the circulation. These changes are invisible to the naked eye, it can be used to measure pulse. Besides motion with low spatial amplitude difficult to distinguish a person may be increased to show the mechanical movement. The basic approach is to consider the time series of color values at all points of the image and exacerbating fluctuations in the band.

Amplification shows the facial flushing caused by blood flow. In this example, temporal filtering is applied to reduce the spatial frequency in order to identify complex distinguishable input signal.

The analysis is based on a linear approximation related to the assumption of a constant stream of image brightness. It allows you to enhance the appearance of movement, do not select the object and determining the movement itself. According to Euler considerations, physical properties of the fluid such as pressure and speed change with time.

Similarly seen a change of pixel values over time. In the Eulerian approach, movements can not gain fixing movement, temporary changes in color enhance certain points. The process of detection and amplification changes is shown in Figure 2.

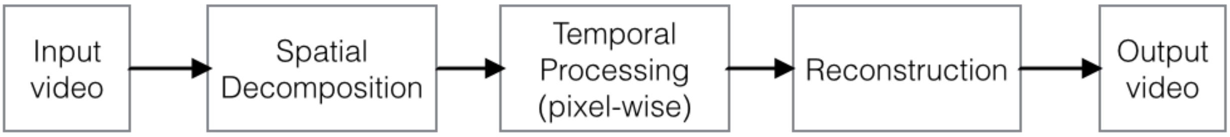
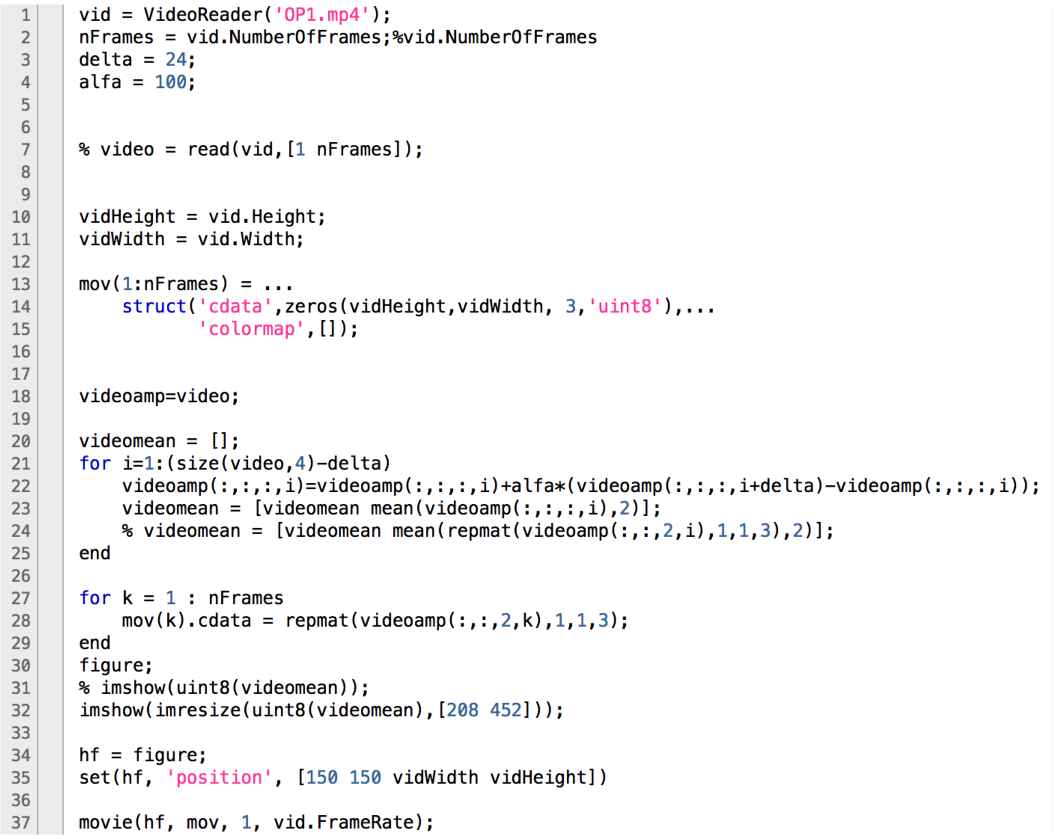


Figure 2: division into bands and signal amplification components.

First video is decomposed into spatial frequency bands. These ranges can be reinforced in different ways because (a) they may have different signal to noise ratio (b) may comprise the spatial frequency for which the linear approximation is impossible. In the latter case, reducing the gain for these ranges in order to suppress the influence of artifacts. If the purpose of spatial processing is simply to improve the temporal relationship of signal and noise by combining several pixels using a filter which eliminates the low frequencies and decreases the resolution of the image being processed to speed up the computation. Then, the processing time of each spatial range.

Consider a series of corresponding pixel values in the frequency range and apply a filter to extract the frequency bands of interest. For example, to select the gain pulse frequency within 0.4-4Gts that corresponds to 24-240 beats per minute. One can manage to remove the heart rate, and use a narrow range around this value. In this case, the processing time can be uniform at all spatial levels, and for all pixels in each level. The Matlab code illustrating how online pulse detection can work is presented on the listing 1 below:

Listing 1: filtering the video sample



```
vid = VideoReader('OP1.mp4');
nFrames = vid.NumberOfFrames;%vid.NumberOfFrames
delta = 24;
alfa = 100;


% video = read(vid,[1 nFrames]);


vidHeight = vid.Height;
vidWidth = vid.Width;

mov(1:nFrames) = ...
    struct('cdata',zeros(vidHeight,vidWidth, 3,'uint8'),...
           'colormap',[]);


videoamp=video;

videomean = [];
for i=1:(size(video,4)-delta)
    videoamp(:,:,:,i)=videoamp(:,:,:,i)+alfa*(videoamp(:,:,:,i+delta)-videoamp(:,:,:,i));
    videomean = [videomean mean(videoamp(:,:,:,i),2)];
    % videomean = [videomean mean(repmat(videoamp(:,:,2,i),1,1,3),2)];
end

for k = 1 : nFrames
    mov(k).cdata = repmat(videoamp(:,:,2,k),1,1,3);
end
figure;
% imshow(uint8(videomean));
imshow(imresize(uint8(videomean),[208 452]));

hf = figure;
set(hf, 'position', [150 150 vidWidth vidHeight])

movie(hf, mov, 1, vid.FrameRate);
```

This piece of code reads a video record. In my case it is a 15 seconds sample collected by web camera at daytime to avoid an influence of electric lamp vibrations. According to [21] I can look at the changes of frequency of the green channel of the video only.

According to [20] I used alpha = 100 and delta = 24 as a values for identifying frequency of blood flow over a face. I used this values and applied them as a filter. On Figure 3 you can see a graph of green channel.

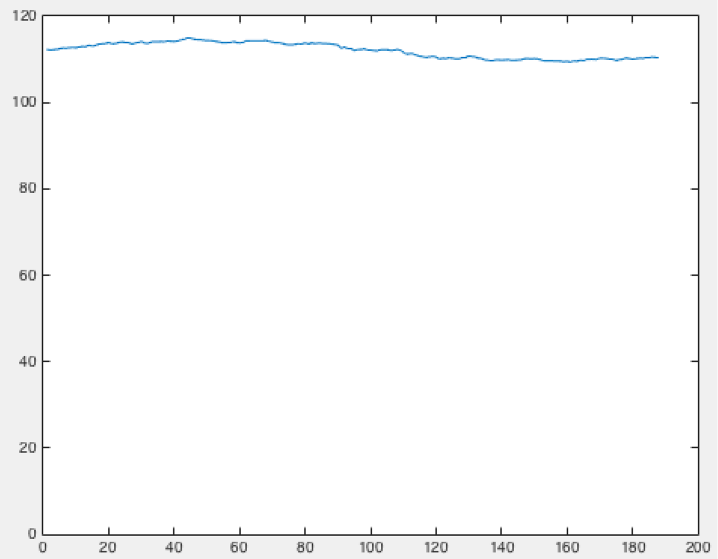


Figure 3: values of the green channel

The next step is applying Fast Fourier Transformation against this signal to find a meaningful harmonic. The results of FFT is presented on Figure 4.

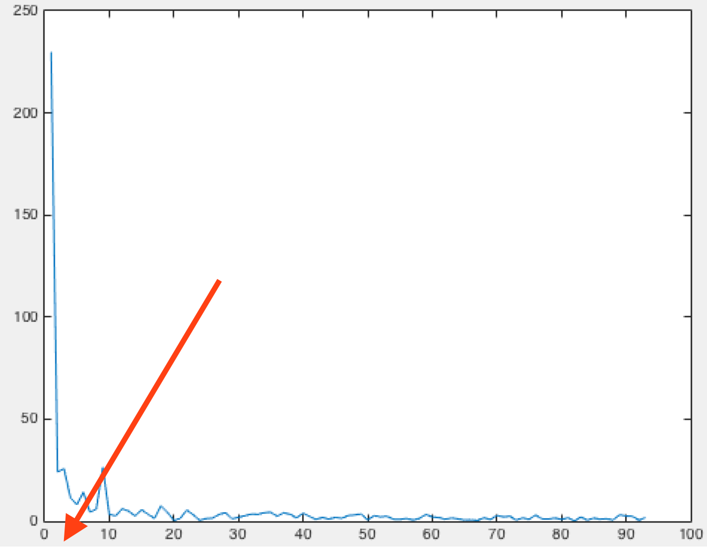


Figure 4: output of FFT

As it is seen on the graph, there is a local maximum on harmonic #8 (pointed by a red arrow). It means that its frequency can be considered as a heart beat frequency. The same results can be observed on figure 5 where a mean value of the image over time is calculated over an source unfiltered image and a filtered image.

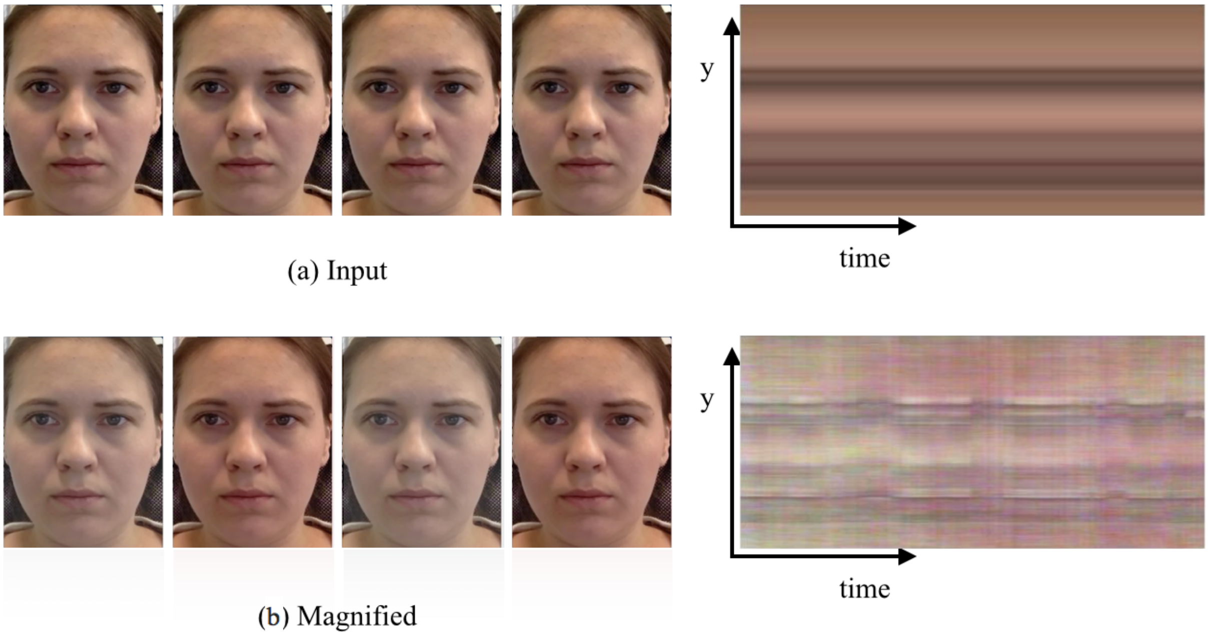


Figure 5: series of input images (a), a series of enhanced image (b) and strip them by appropriate temporal frequencies.

As we can see on the first image the mean image is constant or almost constant meaning that human's eye can't detect the changes of color or tiny movements on the face. The second filtered image has only one change in increased amplitude of of the green channel with delta=24 and alpha=100. We can see that the contrast among the images increased a lot though for absolute values this change is not that significant.

Comparing 2 figures of time series: the first series looks almost as a line an it's impossible to detect a frequency of heart rate over it. On the second series vertical series are easy to detect. Knowing that the length of the fragment was 1 minute and having about 70 vertical lighter lines we can conclude that these lines give us a value of the human's heart rate. [19]

Practice:

In this part I will discuss how to implement video magnification filtering in a web service. The approach continues the same:

(1) select head;

(2) select forehead area to process;

(3) detect frequency of green channel;

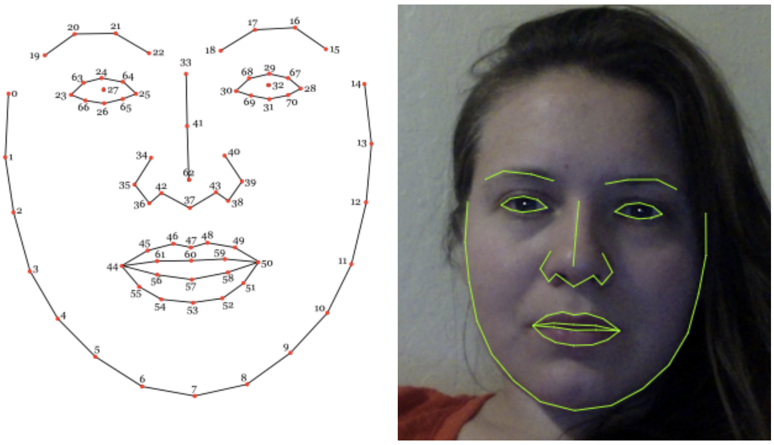
(4) measure the momentum values;

(5) perform Fast Fourier Transformation;

(6) find meaningful harmonic;

(7) give heart rate value.

To perform the first step I'll use clmtrackr [21]. It is a JavaScript library for fitting facial models to faces in videos or images. It reads user's face and builds a series of edge points. An example of clmtrackr work is presented on figure 6.



(a) (b)

Figure 6: a: clmtrackr scheme of end points. b: application of scheme on user's face

It can find 70 edge points. It is the best way to detect certain part of the face with their coordinates. The idea that this points are rebuilt every second so the developer can not only detect certain areas but also monitor them and update the coordinates of the points dynamically. To locate the forehead I used clmtrackr. I assumed that the area to monitor on the human's forehead is a rectangle between eyebrows. I selected the area bounded by eyebrows and 100 px of height. It’s projection is presented on figure 7.

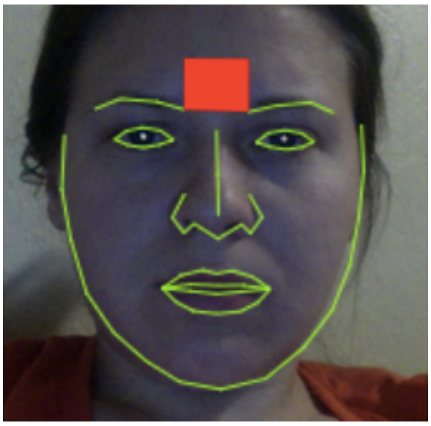


Figure 7: forehead area selected and filled with red using clmtrackr

The next step is to perform Fast Fourier transformation to identify the frequency of the heartbeat. To do this part I used jsfft library [22].

This function the graphs of momentum values of green channel and FFT based on it are performed. On figure 14 the final graphs are presented. The last step to perform is get the value of heart rate based on FFT harmonics. A series of experiments gave me an understanding that harmonic #3 returns the expected value. The last step to perform is calculate Heart rate by knowing the harmonic number and amplitude. To get the heartbeat value the system takes the maximum value of the frequency of harmonic #3, take an average over 1 second and take and average over 15 seconds, dropping every last second. The reliable results appear after the 15th second. The system also allows to write the pulse values to a file for further processing.

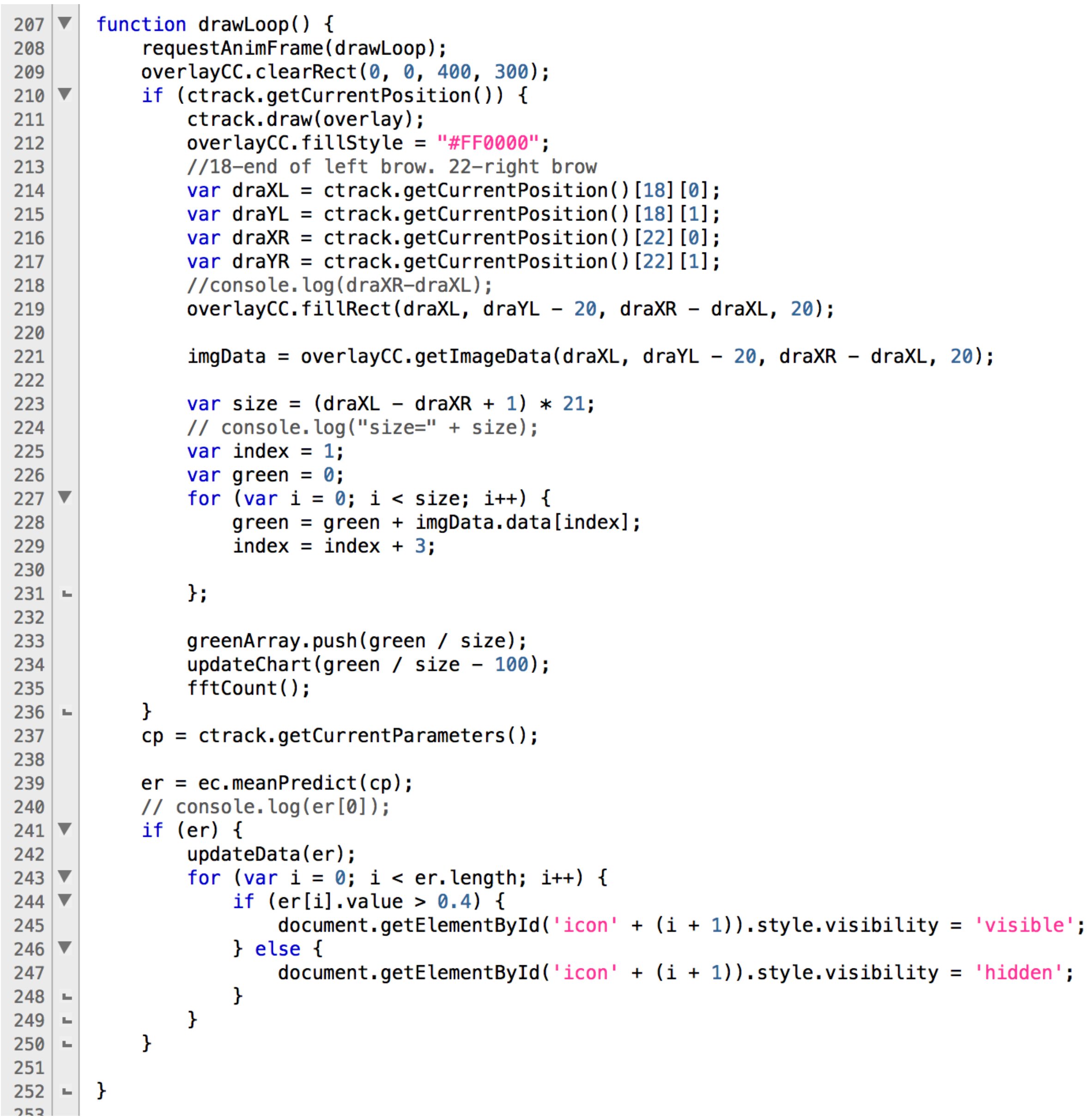
Practical implementation of pulse detector was developed on Javascript is built to locate the face of the user and highlight the forehead area. The system collects time series data to make the simultaneous calculations . It measures mean optical intensity of the forehead area and provides a value of the green channel (the proportion of other channels may be varied to have an optimal value, but the blue channel tends to be very noisy so green-only results were good enough for the implementation).

In optimal conditions and with little noise due to motion, a heartbeat should be sampled in about 15 seconds. Arrhythmic signals should also be visible in the raw data stream.

Once the user's heart rate has been estimated, real-time phase variation associated with this frequency is also computed. This allows for the heartbeat to be exaggerated in the post-process frame rendering, causing the highlighted forehead location to pulse in sync with the user's own heartbeat.

In the listing 2 the function of identifying forehead, taking the value of green area changes and FFT is performed:

Listing 2: draw pulse function drawLoop()



```
function drawLoop() {
    requestAnimFrame(drawLoop);
    overlayCC.clearRect(0, 0, 400, 300);
    if (ctrack.getCurrentPosition()) {
        ctrack.draw(overlay);
        overlayCC.fillStyle = "#FF0000";
        //18-end of left brow. 22-right brow
        var draXL = ctrack.getCurrentPosition()[18][0];
        var draYL = ctrack.getCurrentPosition()[18][1];
        var draXR = ctrack.getCurrentPosition()[22][0];
        var draYR = ctrack.getCurrentPosition()[22][1];
        //console.log(draXR-draXL);
        overlayCC.fillRect(draXL, draYL - 20, draXR - draXL, 20);

        imgData = overlayCC.getImageData(draXL, draYL - 20, draXR - draXL, 20);

        var size = (draXL - draXR + 1) * 21;
        // console.log("size=" + size);
        var index = 1;
        var green = 0;
        for (var i = 0; i < size; i++) {
            green = green + imgData.data[index];
            index = index + 3;

        };

        greenArray.push(green / size);
        updateChart(green / size - 100);
        fftCount();
    }
    cp = ctrack.getCurrentParameters();

    er = ec.meanPredict(cp);
    // console.log(er[0]);
    if (er) {
        updateData(er);
        for (var i = 0; i < er.length; i++) {
            if (er[i].value > 0.4) {
                document.getElementById('icon' + (i + 1)).style.visibility = 'visible';
            } else {
                document.getElementById('icon' + (i + 1)).style.visibility = 'hidden';
            }
        }
    }

}
```

Emotion intensity detection

Theory:

To detect the intensity of the emotions I'll implement a three-step machine learning procedure using Cohn and Kanade's DFAT-504 dataset [18]. The dataset contains 100 college students aging from 18 to 30 years. 65% were women, 15% were Africans, and 3% were Latino. Video samples were take using an analog camera placed directly in front of the subject. Subjects were asked to perform a story of 23 facial emotions. The students started and finished each display with a neutral face. Before showing each emotion, the user described it. Series of images from neutral to all others display were encoded into 640 by 480-pixel files of 8-bit grayscale images.

To perform the learning, testing and validation I selected 313 groups of images out of the dataset because they were labeled and contained all 6 emotions. The samples were taken from 90 students. The first and ending frames (neutral and maximum) were utilized as training images The already trained classifiers were applied to the entire sequence afterward. [7]

The first step of the facial expression recognition is to locate a face on an image. To do it in real time the best approach is using boosting technique. It can be improved with Gentle boost instead of Adaboost, smart feature search, and a cascade training procedure. All these features are implemented in Open source face detection framework [23].

The detector of a face was trained over 5000 samples and millions of non-face samples from an 8000-image dataset. Accuracy on the dataset is 90% false-negative and 1/million false positives, which is good enough.

All faces in the analyzed dataset were detected successfully. The samples were rescaled to 48x48 px. A typical distance between the middles of the eyes was 24 pixels. The images were converted into a Gabor magnitude representation, using a bank of Gabor filters with five spatial frequencies and eight orientations (four at sixteen pixels per cycle). [16]

Gabor filter is a linear filter applicable edge detection in image processing tasks. Gabor filters orientation and frequency representations are likely to human's eye signal capturing and processing so that they can be used for discrimination of images and features. In the complex domain, a 2D Gabor filter can be represented as a sinusoidal wave. [17]

Image analysis using Gabor filters are similar to the perception of the human eye. A sinusoidal wave determines an impulse response (in case of 2D Gabor filters a plane wave) multiplied by a Gaussian function. Because of the multiplication-convolution property, the FFT of a Gabor filter's impulse response is the convolution of the FFT of the harmonic and the FFT of the Gaussian. The filter has a real and an imaginary component representing orthogonal directions. The two components can be added into a complex number or used individually. [16]

Complex:

$$g(x,y;\lambda,\theta,\psi,\sigma,\gamma)=exp\left(-\frac{x'^2+\gamma^2y'^2}{2\sigma}\right)\,exp\left(i\left(2\pi\frac{x'}{\lambda}+\psi\right)\right)$$

Real:

$$g(x,y;\lambda,\theta,\psi,\sigma,\gamma)=exp\left(-\frac{x'^2+\gamma^2y'^2}{2\sigma}\right)\cos\left(i\left(2\pi\frac{x'}{\lambda}+\psi\right)\right)$$

Imaginary:

$$g(x,y;\lambda,\theta,\psi,\sigma,\gamma)=exp\left(-\frac{x'^2+\gamma^2y'^2}{2\sigma}\right)\sin\left(i\left(2\pi\frac{x'}{\lambda}+\psi\right)\right)$$

Where:

$$x'=x\cos\theta+y\sin\theta$$

and

$$y'=-x\sin\theta+y\cos\theta$$

In the equation above, λ stands for the wavelength of the sinus, θ shows the orientation of the vertical to the horizontal stripes of a Gabor function, ψ is a phase, σ is the standard deviation of the Gaussian and γ is the spatial ratio parameter.

The first step of facial expression classification is based on support vector machines (SVM's). SVM's are appropriate to this task because of the high dimensionality of the input data (Gabor representation O(105)) does not harm training time. The training time depends only on the number of learning samples O(102). The tool conducted a 7-way deciding between the following emotion classes: happy, sad, surprised, disgusted, feared, angry, neutral.

In a general case, Support Vector Machines make binary decisions which is selecting in two options. Below the way to use binary classifiers for a multiclass classification is shown. In the given task, the classifier needs to classify six emotions and a neutral state using binary classifiers. The process is done in two steps.On step 1, SVM performs binary decisions. There are three approaches for training binary decisions: one-VS-one, one-VS-all, and all possible combinations. Step 2 translates the features produced by the first stage into a distribution of probabilities over the seven emotion classes. This step can be done using several methods: K-nearest neighbor multinomial logistic ridge regression and a simple voting scheme. [13]

There are many ways of how to downscale a learning algorithm for a multiclass task with a list of binary classifiers. The simplest strategy is to train one versus all. For one versus one distinction, SVM's were trained to classify all emotions pairwise. Classification into seven classes gives birth to 21 SVM. The problem is that in pairwise matching the number of samples for each single SVM can be relatively small and give wrong results. Another way to get rid of this problem is to train a certain emotion against all the rest. Such strategy allows to use more sample images in a training process. To extend one-versus-all approach is also trying all the combinations like one versus six other classes (7 runs), two versus 5 (21 classifier) and three versus four classifiers (35 samples). [13]

II. Combining the results of several binary classifiers. SVM outputs were matched to make a seven options decision. The easiest way to SVM outputs for multiple decisions is voting. The voting procedure calculates the number of classifiers matched with each emotion. If one SVM indicates angry and not sad, angry gets +1 and sad gets -1. These votes added for all of the classifiers. Softmax() makes sure that each class obtained a number between 0 and 1, with a sum over classes.   We matched voting to the nearest neighbor, and to a learned mapping by looking at multinomial logistic ridge regression (MLR). For the nearest neighbor, SVM classifier output for each n of SVM's gives an n-dimensional vector. The checking image is assigned the class of the learning image with the minimal Euclidean distance between their n-dimensional vectors. MLR counts the matrix of weights which maps the results of Stage 1 classifiers onto the seven basic emotions. MLR is a greatest likelihood method, which is similar to a one-layer perceptron with weight decay and with SoftMax [24] competition between the outputs.

For Stage 1, all possible partitions including 2:5 and 3:4 cases gave the best performance while 1 versus all and pairwise partitioning usually performed worse.

SVM performance can be enhanced with Adaboost for emotion classification. AdaBoost is a machine learning meta-algorithm of classification. It is often used together with many other machine learning algorithms to improve the performance. The outputs of any other machine learning algorithms (so called 'weak learners') are added to a weighted sum representing the overall output of the boosted classifier. Adaptiveness of AdaBoost means that a combination of weak learners is shifted to get the value of previously achieved classifier.

$$E_t = \sum_i E[F_{t-1}(x_i) + \alpha_t h(x_i)]$$

It is sensitive to unclear data so the input data should be filtered before classification. The biggest problem with AdaBoost is its trend to over-learning. Though a single classifier can be weak, but while the effectiveness of each of them is better than random guessing (with error rate a little bit better than 0.5), the final classifier can be very strong. [13]

$$F_T(x) = \sum_{t=1}^{T} f_t(x)$$

AdaBoost refers to a particular method of training a boosted classifier. A boost classifier is a classifier in the form where each f_t is a weak learner that takes an object *x* as input and returns a real valued result indicating the class of the object. The sign of the weak learner output identifies the predicted object class and the absolute value gives the confidence in that classification. Similarly, the *T*-layer classifier will be positive if the sample is believed to be in the positive class and negative otherwise.

Each weak learner produces an output, hypothesis $h(x_i)$, for each sample in the training set. At each iteration *t*, a weak learner is selected and assigned a coefficient α_t such that the sum training error E_t of the resulting *t*-stage boost classifier is minimized.

Here $F_{\{t-1\}}(x)$ is the boosted classifier that has been built up to the previous stage of training, $E(F)$ is some error function and $f_t(x) = \alpha_t\ h(x)$ is the weak learner that is being considered for addition to the final classifier.

At each iteration of the training process, a weight is assigned to each sample in the training set equal to the current error $E(F_{t-1}(x_i))$ on that sample. These weights can be used to inform the training of the weak learner, for instance, decision trees can be grown that favor splitting sets of samples with high weights. [13]

The features used for the Adaboost classifier were the Gabor filters. The comparison was performed using 48x48 pixel image samples at five spatial scales. Overall it gave 5x8x48x48=92,160 possible features. To start AdaBoost, a subset of those features should be selected. On each training cycle step, the Gabor filter output with the best classification performance for the current distribution was chosen. The performance metric was a sum of errors on a binary classification task, where the weighting distribution was updated every step to assess how well each training array was already classified.

The work difference in approach for all three algorithms is presented on Figure 8. There the classification steps are illustrated on 92,160 of possible features. [7]

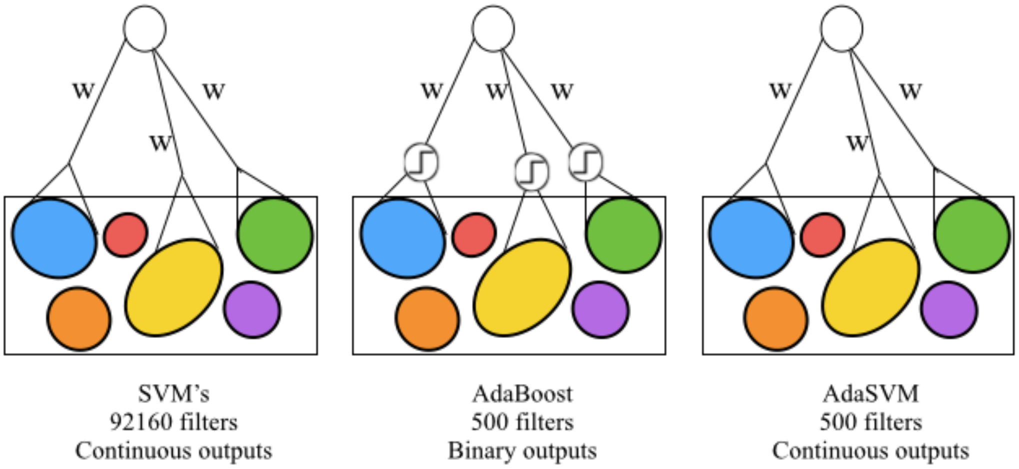


Figure 8: SVM's learn weights for the continuous outputs of all 92160 Gabor filters.

Adaboost training continued until the classifier output distributions for the samples were 100% separated by a gap proportional to the widths of the two distributions. The combination of all features chosen for each of the seven classifiers resulted in a total of 538 features. Classification results are presented in Table 1. [6]

Table 1: leave-one-out generalization performance of AdaBoost, SVM's and AdaSVM's (48*48 images). ω: Gabor wavelength range

ω	kernel	Adaboost	SVM	AdaSVM
4:16	Linear	87.3	86.3	88.9
4:16	RBF		88.1	90.8
2:32	Linear	90.1	88.1	93.4
2:32	RBF		89.1	93.4

The generalization performance with Adaboost was almost the same as linear SVM performance. Adaboost had a substantial speed advantage, as shown in Table 2. There was a 170-fold reduction in the number of Gabor filters used. The convolutions were calculated in pixel space, rather than Fourier space that reduced the advantage of features [6]

Table 2: processing time and memory considerations. Time t' includes the extra time to calculate the outputs of the 538 Gabor filters in pixel space for Adaboost and AdaSVM, rather than the full FFT employed by SVM's

	SVM		Adaboost	AdaSVM	
	Linear	RBF		Linear	RBF
Time t	t	90t	0.01t	0.01t	0.0125t
Time t'	t	90t	0.16t	0.16t	0.2t
Memoty	m	90m	3m	3m	3.3m

Besides being a fast classifier, Adaboost is also a feature choosing technique. Feature selection by Adaboost ensures the researcher that the newly selected contingent on the features that have already been selected. In feature picking by Adaboost, each Gabor filter is a considered to be a weak classifier. Adaboost chooses the best such classifier, and later boosts the weights to decrease the error.

The Gabor features selected by AdaBoost provide one indication of the spatial frequencies that are important for this task. Figure 3 shows the number of chosen features at each of the five wavelengths used. Examination of this frequency distribution suggested that a wider range of spatial frequencies, particularly in the high spatial frequencies, could potentially improve performance. Indeed, by increasing from five to nine spatial frequencies, a performance of the AdaSVM improved. At this spatial frequency range, the performance advantage of AdaSVM's was greater. AdaSVM's outperformed both AdaBoost and SVM's. Moreover, as the input size increases, the speed advantage of AdaSVM's becomes even more apparent. The full Gabor representation was seven times larger than before, whereas the number of Gabor's selected by Adaboost only increased. In respect to this observations, most of the modern facial expression recognition libraries use AdaSVM.

The last thing to determine is an optimal number of Support Vectors. A smaller numbers of support vectors has two basic advantages:

1. the classification procedure is faster,

2. the expected generalization error decreases as the number of support vectors decreases.

The number of support vectors for the linear SVM ranged from 10 to 33 percent of the total number of training vectors. Nonlinear SVM's employed 14 to 43 percent.

The most obvious way to get a series of images to analyze is taking a video record and later split it into a series of images with given frequency. In all of the popular implementations face detection operates at 24 frames/second in 640x480 images. The recognition of facial expression takes from 5 to 20 msec.

In the process each individual image is processed in its own thread and classified, so that the outputs change smoothly like a function of time, especially under illumination and background conditions.

In order to independently encode the specifics of facial expressions, physiologists developed objective coding standards. The facial action coding system is the most popular coding system in the behavioral researchers. A human coder divides facial expressions into a vocabulary of 46 component moving actions. A longstanding research direction in the Machine Perception Laboratory is to recognize facial actions automatically.

This system can be applied to the problem of online facial expression detection. The machine learning methods discussed above were reapplied to the situation where facial action tags changed to basic emotion labels. Face samples were detected and recognized automatically in the video frames.

This system can be applied to the problem of online facial expression detection. The machine learning methods discussed above were reapplied to the situation where facial action tags changed to basic emotion labels. Face samples were detected and recognized automatically in the video frames.

The system was trained again on Cohn and Kanade's DFAT-504 dataset [18] which contains FACS scores [12] by two automatic FACS coders adding to the basic emotion labels that the students put by themselves. Positive samples consisted of the maximum-valued frame of each series of samples, and negative samples consisted of all maximum frames for all the emotions except the expressed one. Besides that the negative sample also contained 313 neutral images collected from the first frames. A non-linear basis function was implemented. Generalization to the newly detected subjects was tested using leave-one-out cross-validation. The results are shown in Table 3.

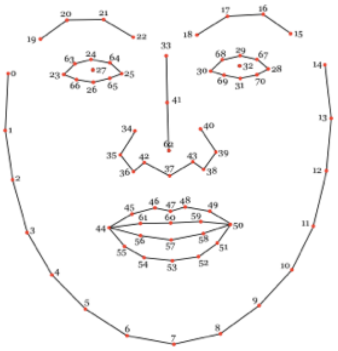
Table 3: generalization results for fully automatic recognition of 7 upper facial actions

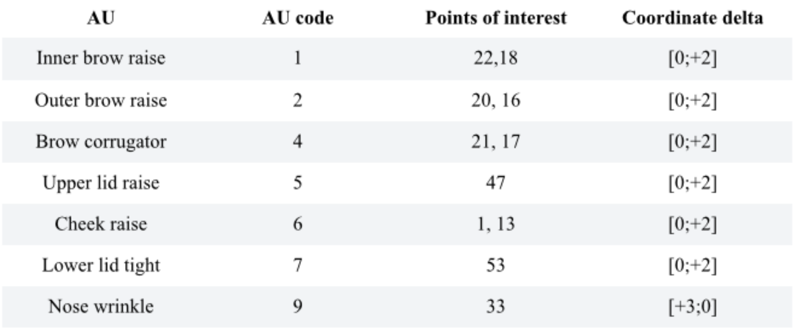
AU	AU code	Agreement	# OF TESTS
Inner brow raise	1	93.6	123
Outer brow raise	2	96.4	83
Brow corrugator	4	89.2	143
Upper lid raise	5	91.8	85
Cheek raise	6	93.8	93
Lower lid tight	7	87.3	85
Nose wrinkle	9	98.6	43

The system reached an average 90% agreement with human FACS labels. This mechanism can be used to detect facial expressions to any source of data quickly and reliably. So this method applies to the online solutions.

Practice:

To implement the emotion detection I used clmtracker [21] with a new classifier trained into 4 classes: angry, sad, surprised, and happy. A significant part of the work was to make the system recognizes FACS elements [12] on a clmtracker face mask. It was done manually. The example is presented on Figure 9.

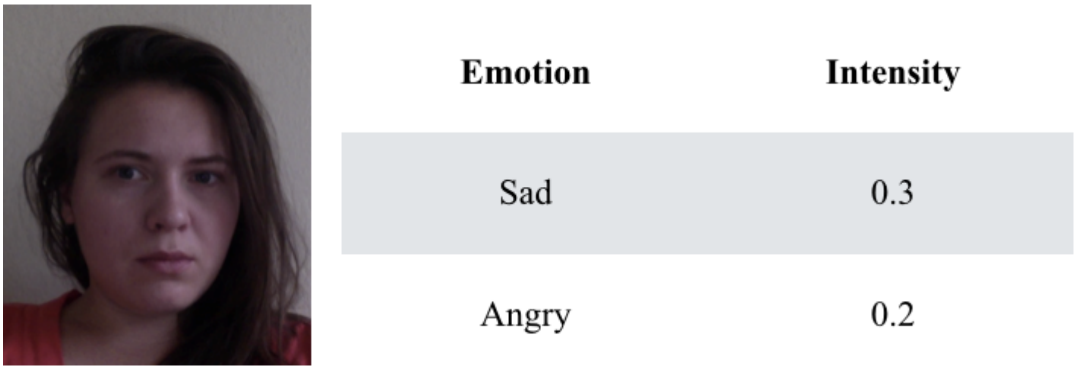




AU	AU code	Points of interest	Coordinate delta
Inner brow raise	1	22,18	[0;+2]
Outer brow raise	2	20, 16	[0;+2]
Brow corrugator	4	21, 17	[0;+2]
Upper lid raise	5	47	[0;+2]
Cheek raise	6	1, 13	[0;+2]
Lower lid tight	7	53	[0;+2]
Nose wrinkle	9	33	[+3;0]

Figure 9: encoding FACS action units into clmtracker points

It made the classifier work quicker which is critical to the web implementation. The classifier was launched 24 times/second to provide almost simultaneous visible classification. To encode each emotion I output 2 parameters: class value (one of “angry”, “sad”, “surprised”, and “happy”) and a normalized result of voting in the range [0..1]. The system allows to return multiple values of the emotions to the given frame. The example is on the figure 10.



Emotion	Intensity
Sad	0.3
Angry	0.2

Figure 10: online emotion detector work. The system can detect multiple emotions and return their intensity in [0..1] range

The system is implemented on JavaScript. It uses common libraries like JQuery to perform typical functions and mathematic operations [25]. It uses pre-defined trained model of a relatively small size: 4 classes with 25 features each. It allows to process the images on the required speed of 24 frames per second.

The structure the tool is presented in the table 4 where all the files, classes and functions are listed. There I discuss the the functions, their parents and the whole logic of the online implementation.

Table 4: classes of emotion detection tool

FileName	ClassName	Function
emotionModel.js	emotionModel	Collected the classifier parameters for an already trained model to match. This pre-defined values allow to classify quickly as there are only 4 arrays of parameters
emotionClassifier.js	emotion Classifier	Head function that collects of the step of emotion classification tool. It takes a trained model and a frame to detect an emotion. To do it calls a number of other functions
emotionClassifier.js	getEmotions	Returns all available functions as an array
emotionClassifier.js	init	Allocates memory and sets variables with required visibility levels
emotionClassifier.js	getBlanc	Making emotion value and emotion weight =0
emotionClassifier.js	predict	Getting the values of the newly classified image
emotionClassifier.js	meanPredict	Correcting the value by looking at other emotions to make sure that all of them return 1 as a sum
fft.js	complexArray	Present the image as a complex function for further spatial processing
fft.js	gabor	Downsample the image and put it into a Gabor filter
stats.js	classify	Performing the classification of outputs of the Gabor filter
utils.js	requestAnimFrame	Open a video frame and show momentum classified results in it
utils.js	initCamera	Make an iframe to open a web camera in the browser and provide access to captured video

Correlation between pulse and emotion intensity

The last step of the project is data analysis and search for correlation between the intensity of emotions and heart rate. To do it I've build a script that showed a stimulus video. For testing I took an advertising clip from Super Bowl [26]. I cropped it into a 1 minute representative sample. While the video was shown I launched the script that turn the web camera on, took the heart rate counted as a frequency of color changes of the green channel of the camera output and measured maximum emotion intensity as a weight of voting mechanism over 4 emotions: angry, sad, surprised, and happy. The sample was collected over 10 respondents with the same testing video, the same camera characteristics on a daylight time to exclude an error caused by the electric light vibrations. The results were collected from Google Chrome log manually and added to MatLab for further processing.

On the first step I collected the samples. One sample is a series of 1,440 records on average taken from one respondent while he was looking Each sample looks like a container of the type shown in the table 5.

Table 5: format of the record

Variable	Sample Value	Units
angry	0.06246441664648956	N/A, [0..1] range
sad	0.20271816598812922	N/A, [0..1] range
surpraised	0.0345414583730949	N/A, [0..1] range
happy	0.09385733920328883	N/A, [0..1] range
timestamp	100.9210000038147	seconds since the program launched
pulse	55.5	bites per minute

A number of statistical parameters are presented in Table 6. The dataset contains the length of the sample in seconds [c2], number of records to analyze [c3], most frequent emotion [c4], its maximum value [c5], an absolute maximum value [c6] and its emotion [c7]. It also contain the maximum [c8], mean [c9] and most popular values of the pulse [c10]. All the calculations are made in Matlab using its statical and matrix operating functions.

Table 6: preliminary statistics about the results of the launches.

#	c2	c3	c4	c5	c6	c7	c8	c9	c10
1	67	1596	surp	0.1097	sad	0.3555	96	69	66
2	66	1577	happy	0.4397	happy	0.4397	135	52	45
3	59	1388	surp	0.2154	happy	0.4950	125	83	78
4	36	854	surp	0.3150	happy	0.7095	127	81	70
5	61	1475	surp	0.0876	happy	0.9554	115	60	45
6	55	1319	surp	0.3512	happy	0.4409	88	51	49
7	61	1505	surp	0.2543	happy	0.5632	112	75	75
8	62	1502	surp	0.0936	happy	0.5770	121	75	45
9	60	1431	surp	0.2694	happy	0.8200	125	87	103
10	59	1459	surp	0.0963	happy	0.6731	135	57	57

Based on the data we can say that in the given video sample the most strong emotion was happiness and the most frequent emotion was surprise though its value was not very significant. This big number of small-weighted surprise patterns may be caused the problems in FACS - clmtracker transformation where I encoded the facial action units with changes of coordinates of the edge points of the face. This issue requires testing on other datasets to decide whether the encoding was incorrect or inappropriate. I can also suggest that the content of the image was entertaining and it was the marketers' idea to make the viewers feel happy and slightly surprised. In this case they completely achieved this goal.

The pulse rate results may look controversial from the first sight. The problem is that reliable information was possible to collect from the 15th second according to [27]. The reason is that FFT requires a bigger sample to make a reliable transformation and decoding. On Figure 11 a pulse of sample 1 is presented. The noise in the first part of the sample is visible.

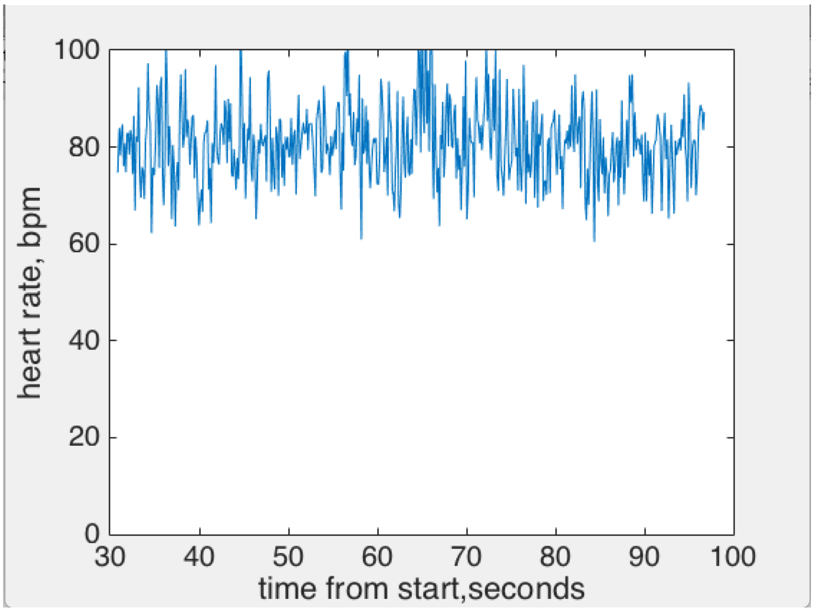


Figure 11: unreasonably high values of heart rate in the first 15 seconds of the sample

Keeping this in mind for the correlation analysis I looked not on the absolute values of the heart rate put on the delta between two neighbor records. I've also dropped the first 15 seconds of the sample not to confuse the results. On Figure 12 changing of pulse and detected raise of emotions is presented.

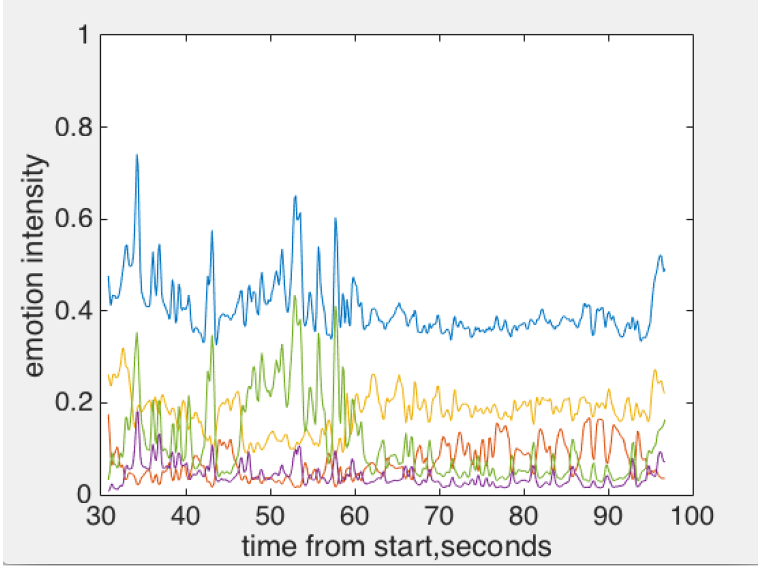


Figure 12: the results of modeling of sample 2

On the axis X the number of records are displayed. On axis Y the value of weighted emotion intensity is shown. Green vertical lines correspond to the records where a an increase of hear rate was detected. Red, blue, purple and yellow lines correspond to happy, sad, surprised and angry respectively. A rise of happy and surprised classes along with the heart rate increasings is easy to detect.

More precise look showed that one of the emotions can affect on the pulse results. In the sample 3 shows (Figure 13) that the rise of heart rate correlates with the growth of the happiness.

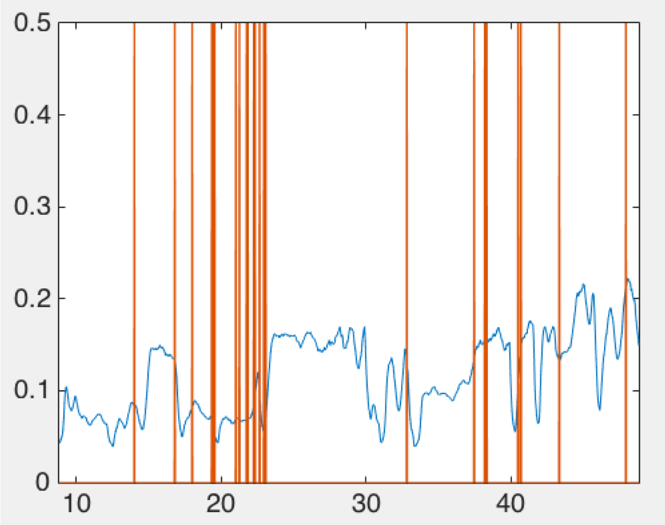
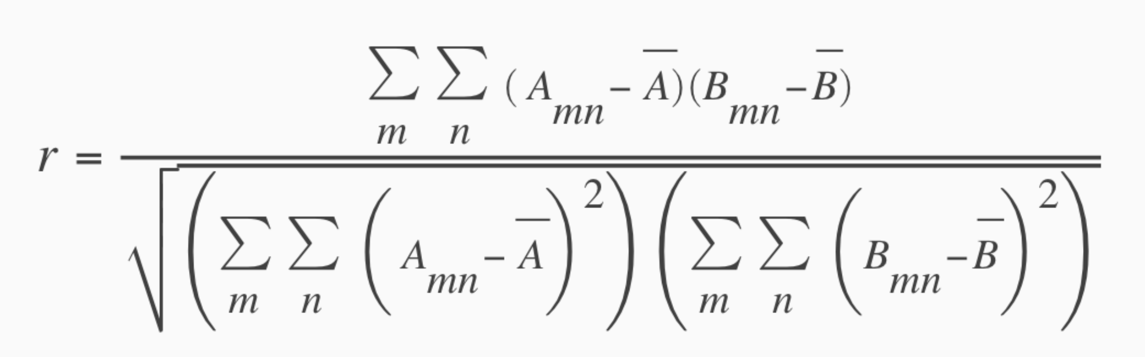


Figure 13: the results of modeling of sample 3

To check this assumption the correlation matrix was built. It shows the results of corr2 Matlab function. This function counts a 2-D correlation coefficient using:



$$r = \frac{\sum_m \sum_n (A_{mn} - \bar{A})(B_{mn} - \bar{B})}{\sqrt{\left(\sum_m \sum_n \left(A_{mn} - \bar{A}\right)^2\right)\left(\sum_m \sum_n \left(B_{mn} - \bar{B}\right)^2\right)}}$$

The correlation between the following arrays is calculated:

[heart rate, heart rate growth] and [weight of each emotion, weight of the sum of emotions]. The results are presented in Table 7.

Table 7: correlation between heart rate and emotions

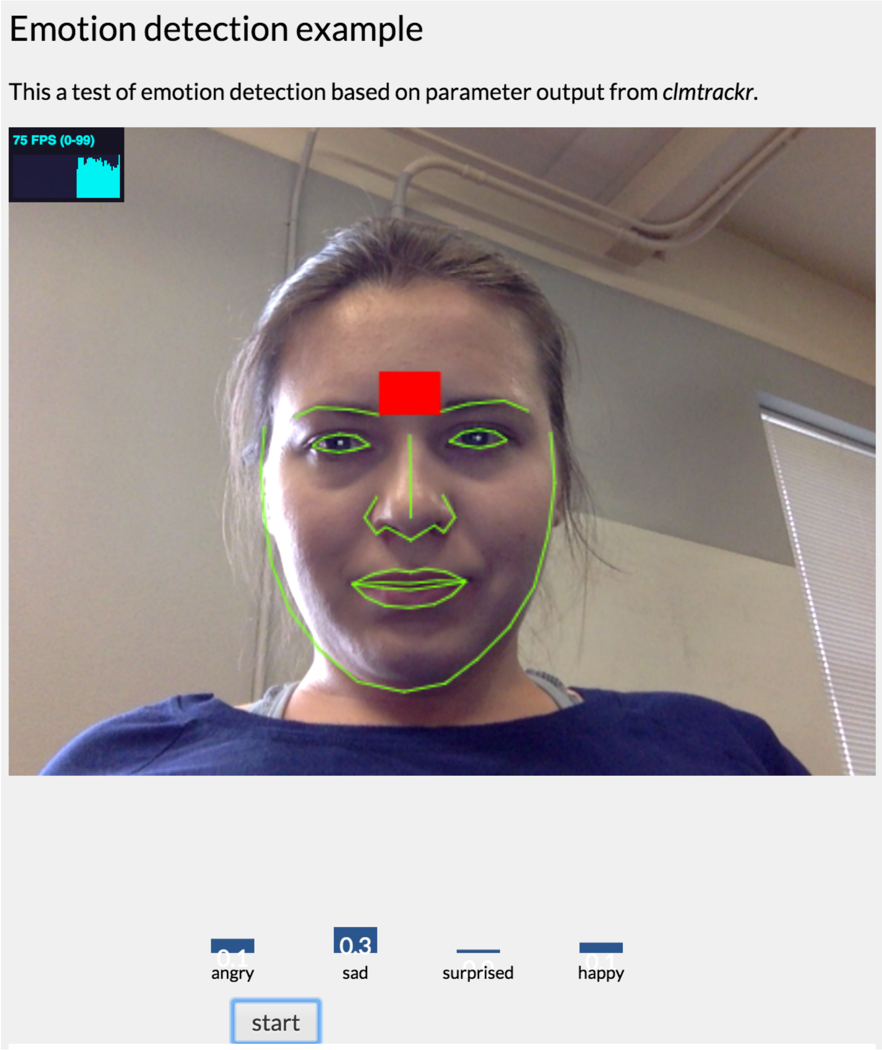
Sample	Parameter	Angry	Sad	Surprised	Happy	Sum
1	hRate	-0.0877	-0.0855	0.0816	0.1325	-0.0234
	hGrowth	0.2118	-0.3165	-0.2039	-0.204	-0.3578
2	hRate	-0.1097	0.2798	0.1748	0.0755	0.3090
	hGrowth	0.2013	-0.1084	-0.1529	-0.3221	0.3429
3	hRate	-0.0875	-0.1124	0.1062	0.0652	-0.0257
	hGrowth	0.3025	0.2042	-0.2023	0.186	0.2875
4	hRate	-0.0690	-0.0372	0.0669	0.2655	0.2966
	hGrowth	0.0060	-0.0023	-0.0041	-0.0121	-0.1763
5	hRate	-0.0408	0.1151	0.0595	-0.0056	0.0267
	hGrowth	-0.0069	-0.0267	0.0055	0.0251	0.137
6	hRate	-0.2986	-0.0859	0.3046	0.2957	0.3117
	hGrowth	0.0023	-0.0026	-0.0047	0.0115	0.1049

As it is seen in the table, the best correlation results are obtained in case of processing momentum changes of the heart rate and the sum of emotion intensity equals to the sum of weights of each emotion.

Online Implementation

The web version of emotion and pulse detection utility is assessable online at http://earthimages.info/EmotionAndPulse/index.html

On figure 14 the sample UI of the utility is presented. It is tested on Google Chrome only. The website is placed on the Bluehost web hosting with Apache web server version 2.2.29.



Emotion detection example

This a test of emotion detection based on parameter output from *clmtrackr*.

Figure 14: UI of the online system

To get the information the user need to accomplish the following steps:

(1) Open URL http://earthimages.info/EmotionAndPulse/index.html

(2) Allow the utility an access to hardware resources of the computer

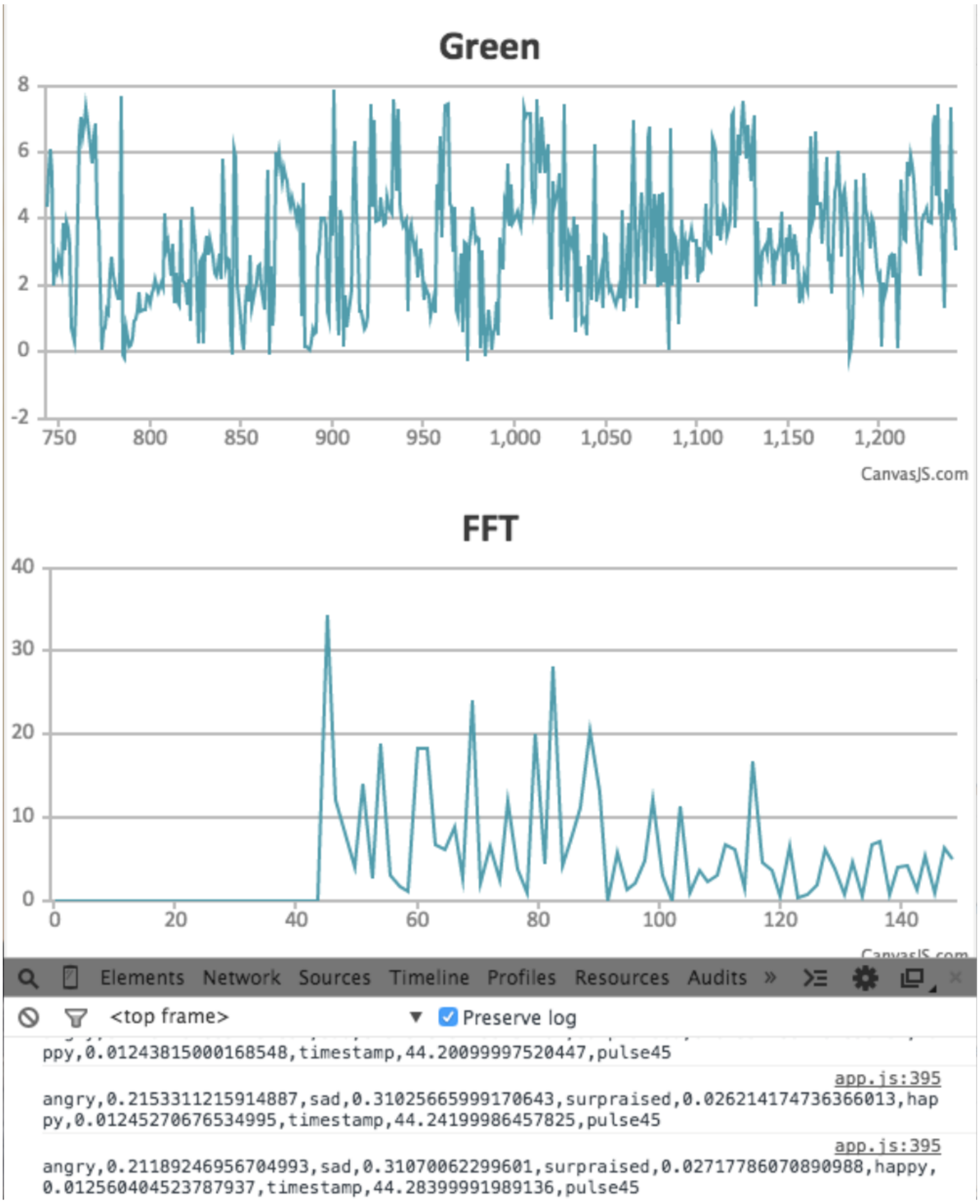
(3) Press start to collect samples for the processing

(4) Wait approximately 100 seconds before FFT has enough records to start building harmonics and extracting frequencies

(5) Open Google Chrome Console output to see all the collected records

(6) Pay no attention to the first 15 seconds of the sample as they are inaccurate

The output of the system looks like presented on figure 15. It returns the graph of momentum values of green channel, output of Fast Fourier Transformation and the records in the Google Chrome console log.



```
ppy,0.01243815000168548,timestamp,44.20099997520447,pulse45
                                                                      app.js:395
angry,0.2153311215914887,sad,0.31025665999170643,surpraised,0.026214174736366013,hap
py,0.01245270676534995,timestamp,44.24199986457825,pulse45
                                                                      app.js:395
angry,0.21189246956704993,sad,0.31070062299601,surpraised,0.02717786070890988,happy,
0.012560404523787937,timestamp,44.28399991989136,pulse45
```

Figure 15: System outputs

The system was built as an minimal viable product of Skoltech-based student project called Brain Selecta [29]. Brain Selecta is a online service that makes surveys based on emotional reactions of human. It can be used for a number of different situations from choosing a profession to A/B testing of commercial advertisements and to assessment the quality of educational materials.

Заключение

Проект был направлен на создание интернет-системы, которая принимает на вход видео записи и выражение лица, интенсивность эмоций и частоту сердечных сокращений по серии снимков и проверяет корреляцию между этими параметрами.

С помощью обзора работ, Open Source сообщества и помощи научного руководителя мне удалось построить такую систему. Для определения необходимых параметров я использовала ряд алгоритмов и методов таких, как AdaBoost, быстрое преобразования Фурье, вычисление корреляции и некоторых других.

Дальнейшие шаги включают проведение большего количества тестов и обучение классификатор на большой группе пользователей, сбор данных для прогнозирования эмоции, основанные только на ЧСС.

Еще одно направление работы включает в себя разработку пользовательского интерфейса, который бы являлся отдельным продуктом. В нем будет можно проследить частоту сердечных сокращений и интенсивность эмоций и прогнозировать эмоции на основании пульса.

Последним направлением дальнейшего развития является интеграция с платформой аналитики медиа-контента e-Contenta, где продукт должен работать в как часть серверного, так что требуется переписать большинство функции на Python или Node.js

Ссылки

[1] Multimodal Emotion Recognition in Response to Videos By: Soleymani, M., M. Pantic, and T. Pun. IEEE Trans. Affective Comput. IEEE Transactions on Affective Computing

[2] Assessing the effects of different multimedia materials on emotions and learning performance for visual and verbal style learners By: Chen, Chih-Ming, and Ying-Chun Sun. Computers & Education

[3] Using emotion recognition technology to assess the effects of different multimedia materials on learning emotion and performanceBy: Chen, Chih-Ming, and Hui-Ping Wang.Library & Information Science Research

[4] FaceFetch: A User Emotion Driven Multimedia Content Recommendation System Based on Facial Expression Recognition By: Mariappan, Mahesh Babu, Myunghoon Suk, and Balakrishnan Prabhakaran. 2012 IEEE International Symposium on Multimedia[5] Emotion detection using sub-image based features through human facial expressions

[6] Dynamics of Facial Expression Extracted Automatically from Video By: Littlewort, G., M.s. Bartlett, I. Fasel, J. Susskind, and J. Movellan. 2004 Conference on Computer Vision and Pattern Recognition Workshop

[7] Fully Automatic Facial Action Recognition in Spontaneous Behavior By: Bartlett, M.s., G. Littlewort, M. Frank, C. Lainscsek, I. Fasel, and J. Movellan. 7th International Conference on Automatic Face and Gesture Recognition (FGR06)

[8] Recent developments in openSMILE, the munich open-source multimedia feature extractor By: Eyben, Florian, Felix Weninger, Florian Gross, and Björn Schuller. Proceedings of the 21st ACM international conference on Multimedia - MM '13

[9] Error Weighted Semi-Coupled Hidden Markov Model for Audio-Visual Emotion Recognition

By: Lin, Jen-Chun, Chung-Hsien Wu, and Wen-Li Wei. IEEE Trans. Multimedia IEEE Transactions on Multimedia

[10] Avec 2013 By: Valstar, Michel, Björn Schuller, Kirsty Smith, Florian Eyben, Bihan Jiang, Sanjay Bilakhia, Sebastian Schnieder, Roddy Cowie, and Maja Pantic. Proceedings of the 3rd ACM international workshop on Audio/visual emotion challenge - AVEC '13

[11] A multimodal fuzzy inference system using a continuous facial expression representation for emotion detection By: Soladié, Catherine, Hanan Salam, Catherine Pelachaud, Nicolas Stoiber, and Renaud Séguier. Proceedings of the 14th ACM international conference on Multimodal interaction - ICMI '12

[12] Meta-Analysis of the First Facial Expression Recognition Challenge By: Valstar, M. F., M. Mehu, Bihan Jiang, M. Pantic, and K. Scherer. IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics) IEEE Trans. Syst., Man, Cybern. B

[13] Generalized Multiclass AdaBoost and Its Applications to Multimedia Classification By: Hao, Wei, and Jiebo Luo. 2006 Conference on Computer Vision and Pattern Recognition Workshop (CVPRW'06)

[14] Theoretical Views of Boosting and Applications By: Schapire, Robert E.. Algorithmic Learning Theory Lecture Notes in Computer Science

[15] Real Time Face Detection and Facial Expression Recognition: Development and Applications to Human Computer Interaction. By: Bartlett, Marian Stewart, Gwen Littlewort, Ian Fasel, and Javier R. Movellan. 2003 Conference on Computer Vision and Pattern Recognition Workshop

[16] Gabor wavelet transform and its application By Wei-lun Chao. http://disp.ee.ntu.edu.tw/~pujols/Gabor%20wavelet%20transform%20and%20its%20application.pdf

[17] Gabor Filter Visualization By V. Shiv Naga Prasad, Justin Domke
http://wwwold.cs.umd.edu/class/spring2005/cmsc838s/assignment-projects/gabor-filter-visualization/report.pdf

[18] The Extended Cohn-Kanade Dataset (CK+): A complete dataset for action unit and emotion-specified expression By: Lucey, Patrick, Jeffrey F. Cohn, Takeo Kanade, Jason Saragih, Zara Ambadar, and Iain Matthews. 2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition - Workshops

[19] Enhanced Eulerian video magnification By: Liu, Le, Le Lu, Jingjing Luo, Jun Zhang, and Xiuhong Chen. 2014 7th International Congress on Image and Signal Processing

[20] Documents Associated With Unified Modeling Language (UML) Version 2.5 - Beta 2 By omg.org http://www.omg.org/spec/UML/2.5/Beta2/

[21] clmtrackr Javascript library by Jason M. Saragih aka auduno https://github.com/auduno/clmtrackr

[22] jsfft Javascript library By Jones N. aka dntj https://github.com/dntj/jsfft

[23] Robust real-time object detection. By Paul Viola and Michael Jones. Technical Report CRL 20001/01, Cambridge Research- Laboratory, 2001.

[24] Softmax Regression http://ufldl.stanford.edu/wiki/index.php/Softmax_Regression

[25] JQuery Javascript library https://jquery.com/

[26] Super Bowl part-time video New 2015 Commercial - #RealStrength Ad | Dove Men+Care https://www.youtube.com/watch?v=QoqWo3SJ73c

[27] webcam-pulse-detector by Hearn, T. aka thearn https://github.com/thearn/webcam-pulse-detector

[28] Deformable Model Fitting by Regularized Landmark Mean-Shift By: Saragih, Jason M., Simon Lucey, and Jeffrey F. Cohn. Int J Comput Vis International Journal of Computer Vision 2010-09-25

[29] Brain Selecta By Tatiana Smirnova http://brainselecta.com

[30] E-contenta https://e-contenta.com/ru/