Human Emotion Based Real-time Memory and Computation Management on Resource-Limited Edge Devices

Yijie Wei, Zhiwei Zhong, Jie Gu
Northwestern University, Evanston, IL
{yijiewei2019, zhiweizhong2021}@u.northwestern.edu, jgu@northwestern.edu

ABSTRACT

Emotional AI or Affective Computing has been projected to grow rapidly in the upcoming years. Despite many existing developments in the application space, there has been a lack of hardware-level exploitation of the user's emotions. In this paper, we propose a deep collaboration between user's affects and the hardware system management on resource-limited edge devices. Based on classification results from efficient affect classifiers on smartphone devices, novel real-time management schemes for memory, and video processing are proposed to improve the energy efficiency of mobile devices. Case studies on H.264 / AVC video playback and Android smartphone usages are provided showing significant power saving of up to 23% and reduction of memory loading of up to 17% using the proposed affect adaptive architecture and system management schemes.

Keywords

Affective Computing; LSTM; System Management; Memory Management; Wearable Devices; Edge Devices.

1 INTRODUCTION

The recent surge in Artificial Intelligence (AI) and Internet-of-Things or the combined Artificial Intelligence of Things (AIoT) has created enormous impacts on nearly all technology sectors. As projected, the deep learning based AI technology is expected to create $15.7 trillion in revenue by 2030, or 14% of global GDP from a study by PwC [1], while the AIoT market is expected to reach $78.3 billion with 39.1% CAGR as reported from [2]. The bloom is partially due to the revolutionary accuracy that deep learning techniques have achieved on image or voice recognition leading to enhanced user experience on mobile applications [3]. While deep learning can address many challenging tasks such as computer vision related classification or natural language processing tasks, there is still one crucial missing link, the "human factor", that has not been well considered or addressed in existing computing techniques and embedded wearable systems.

As a matter of fact, the recent AI computing techniques hold similarities to the human brain for many cognitive tasks. Convolutional neural networks, long-short term memory (LSTM), and deep reinforcement learning all root in the behaviors of the human brain's cortex functions. Therefore, "human-in-the-loop" has become another critical dimension that is being rapidly developed for modern AI technology. In fact, the human factors are already being embedded into many existing AI-related systems showing significant boosts to the targeted AI tasks. For instance, modern recommender systems heavily rely on learning from users' experiences to achieve high prediction accuracy [4]. Smartphones have also been utilized to track users' psychological states to diagnose users' mental health [5]. Recently, Apple has announced a collaborative study on users' depression using smartwatches [6]. In summary, the knowledge of user's real-time feeling and emotions provide incredible opportunities to improve people's life quality.

To address the above demands and opportunities, recently, emotional AI or affective computing has experienced significant growth in research and developments. Emotional AI or "Affective Computing" refers to the computing techniques in both software and hardware that enable detection, classification and utilization of human's affects including both short-term mental states, e.g. stress, excitement, fatigue, or longer-term mental conditions, e.g. depression, personalities, etc. [7]. Recent advancements in embedded devices and wearable technology, e.g. smartwatch, have created new capabilities to bring affective computing into real-life usage [8]. Based on the study from GlobalMarket review [9], the market revenue of affective computing has been projected to grow exponentially from 2016 to 2027 with driving applications including marketing, retail, healthcare, gaming and driving assistance, etc. Essentially, the knowledge of real-time human emotion renders a new dimension and capability to boost the quality of business services and human assistance. While emotion is conventionally considered psychological with subjective bias, significant efforts have been developed to quantitatively and objectively study human emotion. Fig. 1(a) & 1(b) shows the commonly used two dimensional and three dimensional Russell's circumplex model, where special indexes of valence, arousal, dominance are used to quantify people's mental states such as happiness based on the mood angle formed by the three key indexes [10]. Here, valence represents people's "likeness" or "pleasure" dimension. Arousal shows the "activation" or "excitement" of people. Dominance, which is less utilized than valence and arousal, shows people's feelings on "being controlled" or "freedom". Fig. 1(c) shows a recent commercial wristband from Empatica that can be used to collect physiological information from heart rate and skin conductance to derive human emotion [11]. Fig. 1(d) shows an example from MIT business school to detect stress level from skin conductance during exams of students. It shows that strong correlation between mental stress and student's performance exist and the stress measurements from wearable sensors can be used to predict student's performance with 76% accuracy [12].

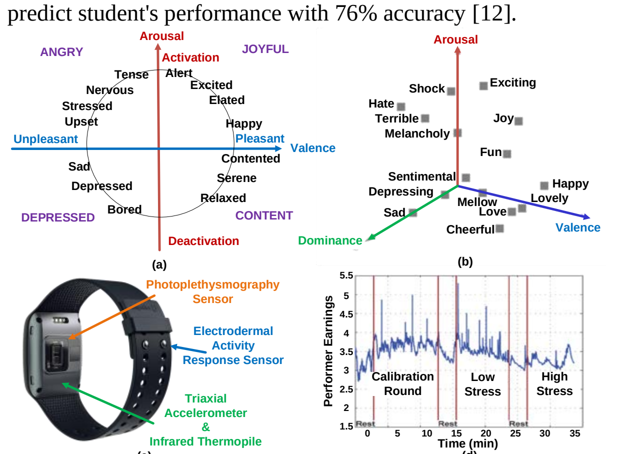


Fig. 1 General Concept of Affective Computing; (a) Two-dimensional Russel circumplex model[10]; (b) Three-dimensional emotion model; (c) E4, Empeca wristband for affect sensing[11], (d) Example of measured skin conductance to derive stress level from students[12].

Existing studies on affective computing mostly focused on two categories (1) detection of human affect based on human

DAC '22, July 10–14, 2022, San Francisco, CA, USA

ACM ISBN 978-1-4503-9142-9/22/07…$15.00
https://doi.org/10.1145/3489517.3530490

physiological signals or activities; (2) applying affect knowledge to improve user's experience or mental/physical health condition. As an example, MIT media labs have shown that driver's stress level can be inferred from heart rate and skin conductance (SC) or galvanic skin response (GSR) sensors, which in turn create an interactive stress management system for a more comfortable and safer driving conditions [13]. The same group also used GSR and accelerometer data to classify people's stress and happiness with 70% accuracy [14]. Improving the workplace condition is another commonly used application for affective computing [15,16]. The recent boom in AR and VR technology for gaming gives another possibility to cooperate with affective computing to change VR game scenes based on players' emotional conditions [17]. In the social media applications, users' eye movements were also tracked to detect users' feelings to improve users' online experiences [18]. In addition, it has been reported that physiological signals can be used to derive user's personality while performing multimedia playback [19]. In summary, many new applications are being explored based on affective computing or "emotional AI" leveraging the advancement of low-power wearable devices and state-of-art machine learning techniques.

However, in all existing developments of affective computing, the focus has been on improving detection accuracy or usage of affects at the system level. There is a lack of study on how the capability of affect detection can be applied at low-level computing architecture or embedded processors to improve the efficiency of computing devices. To close the gap, this work proposes a holistic scheme where the information from affective computing is applied to critical hardware operations such as memory management and multimedia playback. Case studies based on existing datasets are used to demonstrate that the proposed affect driven schemes can significantly benefit the hardware efficiency of low power embedded systems such as smartwatches.

The contribution of this work includes (1) for performing emotion recognition on a resource-constrained wearable device, different deep learning classifiers were compared across different datasets to provide guidance on the model choices; (2) To close the gap between real-time human affects and embedded hardware operation, an affect driven video decoder design and playback scheme is proposed to achieve significant power saving; (3) To further leverage the emotion knowledge of real-time affect detection, an application and memory management scheme on Android smartphone is proposed. As demonstrated in this work, based on the user's affect state, background applications is selectively activated or killed, rendering significant savings on the memory loading effort.

2 AFFECTIVE COMPUTING WITH EFFICIENT CLASSIFIERS

2.1 General System Setup

Fig. 2 shows the general affective computing system setup and application schemes used in this work based on commercial wearable devices such as Apple Watch and smartphones. Since wearable devices and the smartphone are broadly available in daily life, the user's biosignals and daily activity can be easily detected by the smart wearable devices such as smartwatch with built-in physiological sensors such as Photoplethysmography (PPG) sensor, microphone, electrocardiogram (ECG) sensor, and Inertial Measurement Units (IMU). To extend the limited battery life of wearable devices, power-hungry signal processing, feature extraction, and classification work, e.g. deep learning for affect classification, may be handled by more powerful smartphone application processors. For instance, Apple's A13 processor and Qualcomm's Snapdragon 855 are empowered by built-in "neural engine" cores for deep learning tasks.

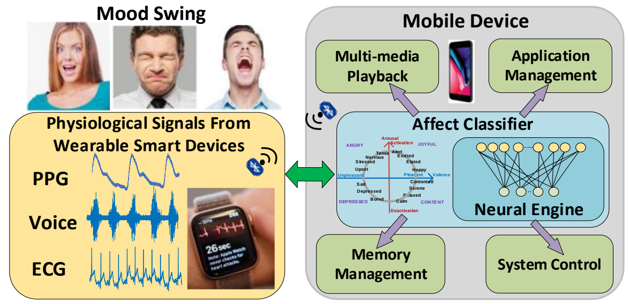


Fig. 2 General system configuration of affective computing in this work.

The availability of "neural engine" on a smartphone and the rapid developments of affective computing opened a door to establish a connection on the user's psychological feeling and the hardware/software operations of embedded devices, which have not been studied previously but hold strong promise to provide a new class of "human machine interface". For instance, an emotion-aware recommendation system can provide users with a much more accurate shopping and online experience. As proposed in this work, an intelligent emotion-based application and memory management system can optimize the user's favorite applications' runtime and background status for enhanced energy efficiency. Hence, this work explores the design and optimization of a wearable embedded device exploiting the use of affective computing.

2.2 Hardware Efficient Affect Classifier

Although machine learning models for emotion classification tasks have been explored in previous works[14], there is a lack of existing work considering real-time emotion classification on wearable devices. To unleash the potential of "real-time" affective computing, embedded hardware deployment of an affect classifier is highly beneficial despite of challenges. Due to the power-hungry operation of the deep learning classifier and limited memory and battery of wearable devices, we herein study the accuracies of different classifiers to guide system design for wearable devices.

Three emotion databases are used in this study with popular classifiers. The RAVDESS dataset [20] contains 7356 sets of emotional speech and songs by 24 actors. The EMOVO dataset [21] includes 14 sentences with different emotions in Italian by six actors. The CREMA-D dataset [22] include 7442 video clips from 91 actors with 12 English sentence. All these datasets are labeled with emotional labels, e.g., neutral, sad, fear, angry, and happy. The input voice files are processed to generate the input features, including Mel-frequency cepstral coefficients (MFCC), zero crossing, root-mean-square deviation (rmse), sound pitch, and magnitude. We study three commonly used classifiers, including the fully connected neural network multi-layer perceptron (MLP) classifier, convolution neural network classifier (CNN), and long-short-term memory (LSTM), which are specialized for time series classification. TensorFlow Keras library is used for building the models. Due to the limitation of wearable devices, small-size models for each classifier are built so that they can be deployed into smartphone or smartwatch devices for real-time detection. The MLP model contains three layers with total 260 neurons and 508 k trainable parameters. The CNN have around 649 k parameters with three layers of 32, 64, 128 neurons on each layer respectively. The

LSTM have two layers with total 320 units and around 429 k trainable variables.

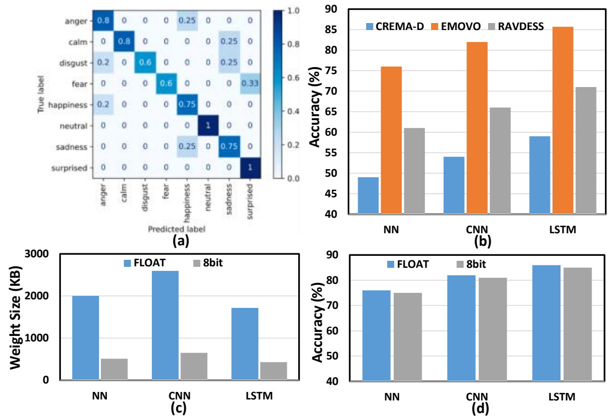


Fig. 3 (a) Confusion matrix of RAVDESS dataset by LSTM classifier; (b) Overall classification accuracy by each model; (c) Weight size with floating point and 8-bit quantization of each model; (d) Accuracy with different precisions of each model.

Fig. 3(a) shows one example of the confusion matrix from the LSTM classifier results on the RAVDESS dataset. Fig. 3(b) shows the overall average classification accuracy among databases and models. Fig. 3(c) shows the weight size of each model based on EMOVO dataset, which highlights the hardware resource needed. Fig. 3(d) shows the quantized 8-bit accuracy on EMOVO dataset of each model with less than 3% accuracy loss versus floating point model. Amount all the results, the CNN and LSTM classifiers show better performance than the MLP classifier. Considering model size and accuracy, LSTM is observed to be more attractive to be deployed on resource limited wearable devices based on the datasets studied herein.

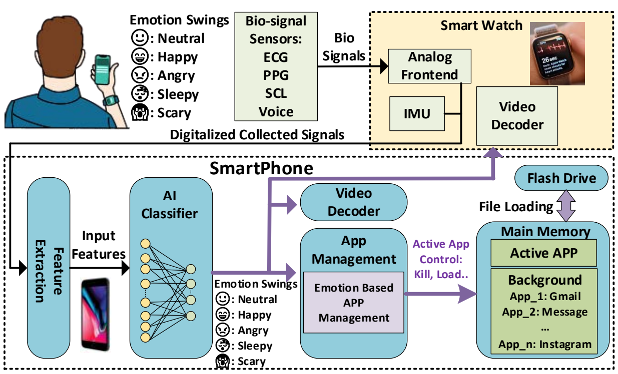


Fig. 4 Proposed system configuration for emotion-driven real-time memory and multimedia playback.

3 AFFECT-DRIVEN ENERGY-EFFICIENT SYSTEM MANAGEMENT

To exploit the knowledge of real-time affect detection from an on-device classifier as discussed in Section 2, in this work, we propose novel system management schemes to improve the energy efficiency of the embedded system. Fig. 4 shows the overall architecture and signal flow of the proposed affect-driven real-time memory/application management and multimedia playback schemes. The smartwatch is used to collect biosignals from the user with external sensors. The biosignals are processed from analog frontend modules and sent back to the user's smartphone for on-device classification. Feature extractions such as frequency-based spectrum diagrams such as MFCC and time-based features such as mean, histogram, and variance are extracted and sent to the classifier. The results from the smartphone's AI classifier, e.g. "neural engine," are used to generate the accurate emotion labels used for the proposed real-time affect-driven video decoder and application management module for efficient system control.

Two schemes are proposed in this work. First, as described in section 4, we propose an adaptive multimedia hardware design and management where the video decoders on wearable devices such as smartwatches can adaptively change the power saving strategy with the user's emotional states to fit the user's requirements. Second, in section 5, we propose an emotion-based smartphone App management scheme that adaptively allocates the limited RAM resources to user emotion related apps to reduce the power and time-costly reloading operations from flash memory.

4 AFFECT-ADAPTIVE MULTIMEDIA MANAGEMENT SCHEME

To improve the power efficiency of video decoders based on users' emotions, we proposed hardware modifications and relevant adaptive control schemes based on the conventional implementation of H.264/AVC video decoders for low power devices. The architecture of the proposed affect-driven H.264/AVC video decoder is shown in Fig. 5. In the conventional H.264 decoder [23], the decode flow starts with a 128-bit Circular Buffer that receives compressed H.264 video bitstream from the source. Bitstream Parser fetches the input video bitstream from the Circular Buffer and processes it with CAVLC Decoder and Variable Length Decoder. IQIT Decoder recovers the residual data in the form of 4x4 blocks. Inter/Intra Prediction modules are used to generate predicted pixels in the form of 4x4 blocks. The sum of residual and predicted data is sent to Deblocking Filter, which operates on the edges of macroblocks (MB) for better video quality. Finally, the decoded video data in YUV format is sent to external displayers.

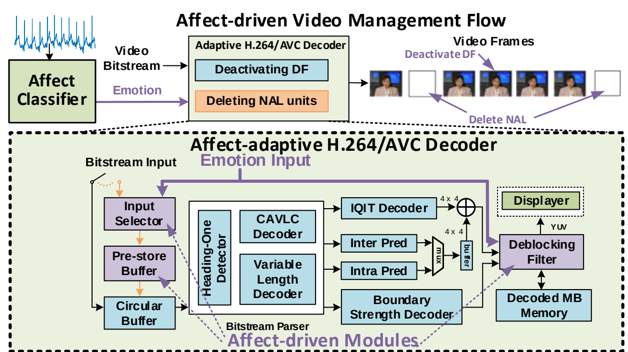


Fig. 5 The system architecture of the proposed affect-driven H.264/AVC video decoder.

For the proposed affect-driven H.264 decoder, two strategies, adaptive deletion of non-critical input bitstream and deactivation of Deblocking Filter (DF), are deployed to save power consumption while slightly reducing decoded video quality in different scales based on the emotional states. The compressed H.264 bitstream goes through Input Selector, which reduces the size of the input bitstream and stores it in an additionally inserted Pre-store Buffer of 128 x 16 bits for the use of emotion adaptation. Once the data is available, the Circular Buffer starts to fetch data from the Pre-store Buffer, and the video decoding proceeds. Emotion Input determines the level of input bitstream reduction, which leads to an adjustable range of power saving since the size of data to be processed is controllable. Typically, a more significant input data reduction results in more power saving but less video quality. The second knob added is the activation of the Deblocking Filter, decided by the emotional states. According to the experimental results, the deactivation of the Deblocking Filter reduces up to 31.4% power consumption with minor degradation of video quality in terms of fuzzy MB edges, with details shown later.

The input data reduction by the Input Selector should be carefully managed to provide a wide range of adjustable video quality and avoid significant quality loss. Inside the compressed H.264 bitstream, three types of frames are associated with video contents: Intra-coded (I) frames, Predicted (P) frames, and Bi-directional predicted (B) frames. I-frames contain the data needed to recover an entire image, which means decoding an I-frame is independent of other frames and is mostly utilized as the reference for decoding other frames. The decoding of P-frames requires decoding of previous images, and B-frames require prior decoding of subsequent images. Data of the three types of frames are packed in Network Abstraction Layer (NAL) units that begin with a start code (i.e., 0x000001 or 0x00000001) and subsequent identification bits for I, P, or B frames. In a group of sequential pictures with a similar background, an I-frame contains all the compressed information for the first picture. It is used as an indispensable reference and needs to be maintained to the largest extent. P-frames and B-frames contain the difference between these pictures and need fewer bits to be encoded. Therefore, to achieve maximum power saving with minimum quality impact, we abandon some data from P-frames and B-frames through the Input Selector, which leads to minor video quality when the emotional data indicates it is not critical.

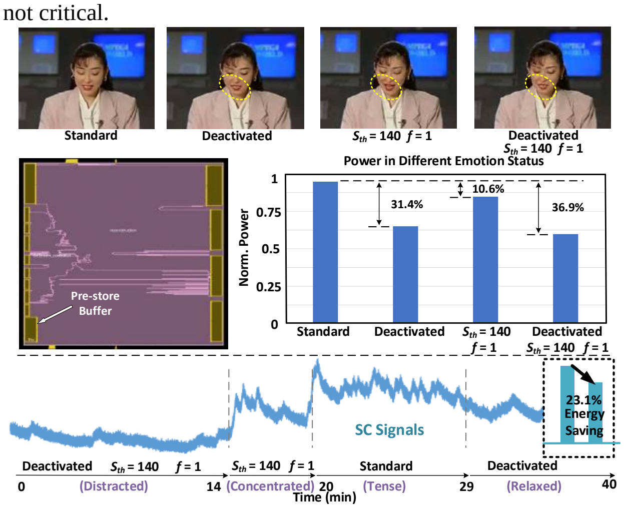
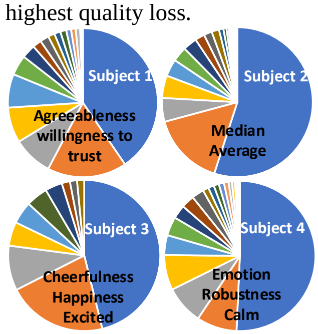


Fig. 6 Video decoder output (top); Video decoder layout and power saving of the affect-driven H.264/AVC video decoder compared with conventional design(middle); Skin conductance (SC) signal from one example in the database[24] and affect-driven video play back energy saving(bottom).

Two parameters are given in the Input Selector when deleting NAL units for B-frames or P-frames to adjust the degree of video quality loss according to the emotion input. The first one, denoted by Sth, stands for the threshold size of NAL units. For example, a NAL unit for P-frame of B-frame is considered fine to be abandoned if its size is smaller than or equal to S_th. The second parameter, denoted by f, indicates how frequently to delete NAL units. Generally, if the input bitstream has n NAL units, the sizes of m NAL units (for P-frames or B-frames) are smaller than or equal to S_th bytes, the number of deleted NAL units will be m/f. Larger Sth stands for larger m, and larger m/f indicates more data is to be abandoned, leading to more power saving but worse video quality, and vice versa. As the input bitstream goes through the Input Selector and stores in the Pre-store Buffer, the category and size of each NAL unit are analyzed. If a NAL unit is decided to be deleted, the Input Selector will adjust the writing address of the Pre-store Buffer to replace the NAL unit to be abandoned. While the Input Selector writes data to the Pre-store Buffer, the Circular Buffer fetches data. Therefore, a hand-shake scheme is established between the Input Selector and the Circular Buffer to avoid read-write conflicts. Overall, in the proposed affect-driven H.264 video decoder, the emotion input decides the degree of video quality loss by tuning values of S_th and f, and deactivating the Deblocking Filter, which makes the power saving adjustable according to the users' emotions.

The proposed H.264/AVC decoder design is implemented using commercial 65-nm CMOS technology with the layout shown in Fig. 6. The decoder has a size of 1.9 mm^2 with a supply voltage of 1.2V and runs at a clock frequency of 28MHz. The introduction of the Pre-store Buffer causes 4.23% area overhead compared with the conventional design. By altering the threshold size S_th, the frequency f of deleting NAL units, and the activation of the Deblocking Filter, four working modes of the video decoder are provided in Fig. 6. The standard mode, meaning all NAL units are processed and the Deblocking Filter is not deactivated, provides the best video quality with the highest power consumption. The deactivation of the Deblocking Filter mode leads to 31.4% power reduction and fuzzy MB edges as shown in the circled part of the corresponding video frame. The deletion mode, with S_th=140 and f=1 as an example, reduces the power consumption by 10.6%, and the output image enjoys a slightly better video quality than that of the deactivation mode. The combination of deactivating the Deblocking Filter and deleting input data yields the most power saving among the four modes (about 36.9%) at the cost of the highest quality loss.



Platform	Android Studio 2021
Emulator Version	Android 11 API 30
CPU CORE	4
Ram Allocation	4096 MB
Rom Allocation	32GB
# of Total Apps	44
Resolution	1920x1080

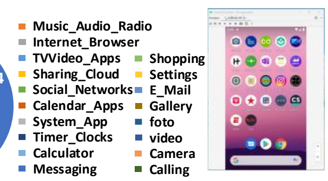


Fig. 7 App usage pattern by category from 4 users with different personality (left). Android emulator specification table with screenshot (right).

To simulate the proposed affect-driven video playback operation, we applied our proposed scheme to uulmMAC database [24]. In [24], the skin conductance (SC) signals were measured on a user when watching a visual-search-task video for 40 minutes and user's emotion is labeled as "relaxed", "distracted", "concentrated", etc. Hence, the magnitude of the varying SC signal could be used to derive users' emotions. While the emotion to video mode control configuration can be programmed based on the user's preference, we made a simplifying assumption in this case: When the user is in distracting status, the quality of the video is not critical, the affect-driven H.264 decoder works in the most power-saving mode with the deactivation of the Deblocking Filter as well as S_th=140 and f=1, namely all the NAL units with size $\leqslant$ 140 bytes are abandoned. As the user becomes concentrated as indicated by the impulse of the SCL signal at the time of 14 minutes in Fig. 6, the Deblocking Filter is turned on, resulting in better video quality and higher power cost. When the user is highly concentrated at the time of 20 minutes, the standard mode is turned on to provide the best video quality with the highest power consumption. When the user is at the relaxed emotion at the time of 29 minutes, the Deblocking Filter is deactivated again to trade video quality for power saving. Compared with displaying the whole video in the standard model, the affect-driven playback of the video saves an overall 23.1% energy consumption. The power adjustment strategy is subjective to the user and hence is expected to be personalized and reprogrammed with the hardware capability provided in this work.

5 AFFECT-DRIVEN APP and MEMORY MANAGEMENT

5.1 Phone App Usage Analysis with Affects and App Background Management Scheme

Due to the lack of an existing affect database with high-level smartphone system usage, we use a personality-based cellphone usage database as an approximation to users' affect related behavior because personality is strongly related to long-term affects. In prior research on the personality and the daily cell phone usage pattern [25], 640 subjects' personalities were evaluated by a score system based on Openness (O), Conscientiousness (C), Extraversion (E), Agreeableness (A), and Emotional Stability(ES) and subjects' daily cellphone usage were recorded. In this work, we randomly picked 4 subjects for study. The distribution of subjects' top 20 daily used apps is shown in Fig. 7. Among the four subjects, messaging and internet browsing dominate the daily app usage with about 60% to 70% in total. The rest 30% to 40% app usage shows a large variation among the four subjects depending on their personality. The first subject with high score of "agreeableness and willingness to trust" accesses the radio, sharing cloud and TV video apps frequently. The second subject has a moderate personality with median scores from the score matrix has the same amount of app usage on sharing clouds, internet browsing, and tv video Apps. The third subject with high "cheerfulness and positive mood" scores has more App usage on calling and shared transportation. The fourth subject with median scores also has very even app usage patterns. In this work, we use different subject's personality to emulate the impact of different affects to the user's App usage patterns.

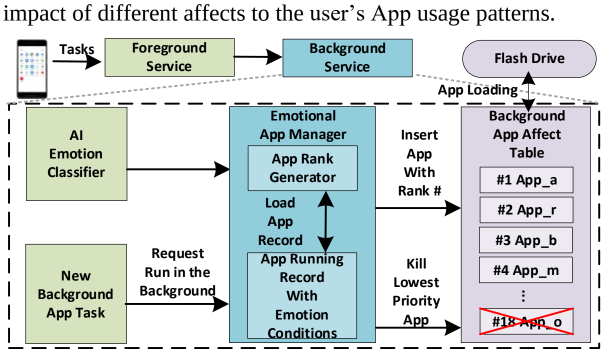


Fig. 8 Emotion adaptive App background management flow.

In modern OS of smartphones, e.g. Android, the smartphone App manager keeps frequently used app running or cached within the memory to save the loading energy and startup time. For the apps with a lower chance of running, the OS will keep the Apps until the new app is being used and the memory or background process limit is reached. Since the activation possibilities of running Apps have a high correlation to the person's emotional states, we explore an affect-driven strategy where the smartphone keeps the most possible Apps in the background according to the detected user emotion states and kill the unlikely Apps when the process limit or memory was reached leading to saving of the memory fetch or reloading efforts from flash drive to the memory.

Fig. 8 shows the proposed emotion or affect adaptive app background management flow. In a typical Android App management scheme, the foreground service module is responsible for running the Apps with user noticeable operations like notifications. The background service module manages the background App activities. The proposed affect-adaptive App management scheme is working with the background manager to decide whether to keep the App in background or remove it from the list when the number of background Apps is over the process limit (e.g. typically 20 in Android OS) or the free memory space has run out. The emotional background manager has an App rank generator and a background "App Affect Table". The App Affect Table stores the the user specific app usage pattern with certain emotional states. To save the power cost by loading Apps from flash drive, the affect-adaptive App manager provides the more likely Apps higher priority based on the detected user's emotion. When the emotion changes, the preferred Apps based on the new emotion state will be given a higher priority and push other Apps to a lower rank. Compared with the first-in-first-out strategy in the basic Android system, this method saves significant memory loading power and time on the frequently used Apps.

5.2 Experimental Results

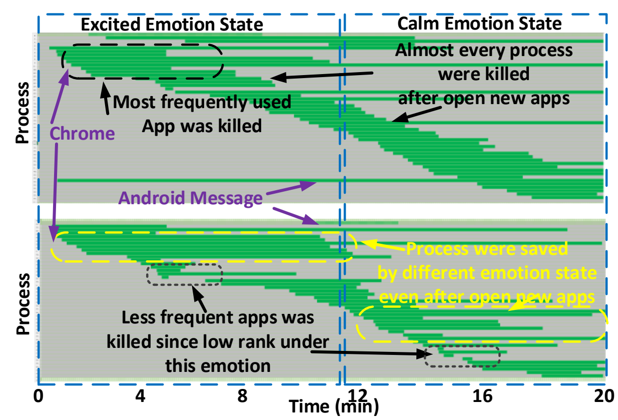


Fig. 9 Process running diagram in exciting emotion state and calm emotional state with system default baseline scheme (top) and proposed scheme(bottom).

Fig.7 (right) shows our experimental setup using an Android Emulator with 4 GB memory on Android 11. A total of 44 Apps selected from the database were installed on the emulator to represent the daily used App from each category reported in the previous study [25]. A monkey script was used to simulate the user's daily usage pattern to open certain Apps with a given frequency and duration to match the probability of the subjects' daily statistics of App usage as in Fig. 7. The monkey script also generates random touch and typing inputs during the interaction with each Apps. Meanwhile, the Android system developer tracing tool traced the CPU and memory usage pattern which are recovered by Perfetto developer API for data analysis. To reduce the simulation duration, we shortened the operation time of each app and remove the idle time of the users, which will not have significant impact to the results since our focus is on the loading activities of each new App being activated.

Fig. 9 shows the simulated user daily App usage pattern. The pattern are generated from subject 3 and 4 who represent excited mood and calm moods. The horizontal green line shows each application process' lifespan during the simulation, while the grey line represents that the process is not running or being killed. The first 12 min stands for the user are in excited emotion state and followed by a 8 mintues calm emotion state. In the original default setting(top), the system follows mostly a first-in-first-out killing strategy when new Apps kick in and reach the process limit (default limit at 20) or memory limit. The exceptions are system Apps and frequently used processes of the users, such as Android messages, which are never killed due to the periodic usage. The bottom figure shows the affect-driven App management where the most likely Apps related to the exciting emotion were kept in the memory. The unlikely used Apps based on the users' emotional usage pattern was killed soon after new Apps kicked in to release memory. After the emotional state changes, the killing priority of Apps is also adjusted according to the emotional usage pattern. The Apps unrelated to the emotional state were likely being killed to release memory spaces.

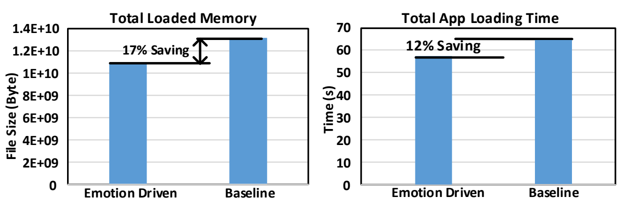


Fig. 10 Total memory loaded at App start time (left) and App loading time (right) in the case study of this work.

Fig. 10 shows the total amount of loaded memory at App start time and loading time by the selected Apps during the simulated affect-related App usage sequence shown in Fig.9. In the proposed emotion driven management case, the memory loading saving comes from roughly equal saving of file loading from flash drive and app-specific allocated memory space. As shown in the figure, the proposed scheme achieves 17% saving of total memory loaded at App start, and 12% saving of loading time compared to the system default background management scheme.

6 CONCLUSION

The real-time emotion based affective computing offers an unprecedented path to tighten the connection between human users and edge devices. In this paper, we explore a deep collaboration between human emotions and embedded hardware management to achieve enhanced computing efficiency on resource-limited edge devices. Different machine learning classifiers were studied to guide the model selection on resource limited edge devices. An affect driven hardware video decoder was proposed achieving up to 23% power saving following the proposed adaptive scheme. An affect driven application and memory management scheme on the Android operating system was also proposed with up to 17% savings on memory loading thanks to the system adaptation to the real-time user's affect.

ACKNOWLEDGEMENT

This work was supported in part by the National Science Foundation under grant number CNS-1816870.

REFERENCES

[1] PWC, "Sizing the prize What's the real value of AI for your business and how can you capitalise?" [Online].

[2] "Artificial Intelligence of Things Solutions by AIoT Market Applications and Services and Industry Verticals 2021 - 2026." [Online].

[3] "Neural Networks And Machine Learning Are Powering A New Era Of Perceptive Intelligence." [Online].

[4] H.-T. Cheng *et al.*, "Wide & Deep Learning for Recommender Systems," *ArXiv160607792 Cs Stat*, Jun. 2016, Accessed: Nov. 20, 2021. [Online].

[5] H. Sarker *et al.*, "Finding Significant Stress Episodes in a Discontinuous Time Series of Rapidly Varying Mobile Sensor Data," in *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*, San Jose California USA, May 2016, pp. 4489–4501. doi: 10.1145/2858036.2858218.

[6] "Online Resource: Apple Is Working on iPhone Features to Help Detect Depression, Cognitive Decline." [Online].

[7] J. Tao and T. Tan, "Affective Computing: A Review," in *Affective Computing and Intelligent Interaction*, vol. 3784, J. Tao, T. Tan, and R. W. Picard, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2005, pp. 981–995.

[8] L. Shu *et al.*, "A Review of Emotion Recognition Using Physiological Signals," *Sensors*, vol. 18, no. 7, p. 2074, Jun. 2018, doi: 10.3390/s18072074.

[9] "Affective Computing Market Size, Share & Trends Analysis Report By Technology(Touch-based, Touchless), By Software, By Hardware, By End-use (HealthcareAutomotive), And Segment Forecasts, 2020 - 2027." [Online].

[10] J. Posner, J. A. Russell, and B. S. Peterson, "The circumplex model of affect: An integrative approach to affective neuroscience, cognitive development, and psychopathology," *Dev. Psychopathol.*, vol. 17, no. 03, Sep. 2005.

[11] "Embatica website." [Online]. Available: https://store.empatica.com/products/e4-wristband?variant=17719039950953

[12] C. Mundell, J. P. Vielma, and T. Zaman, "Predicting Performance Under Stressful Conditions Using Galvanic Skin Response," *ArXiv160601836 Cs Stat*, Jun. 2016, Accessed: Nov. 14, 2021. [Online].

[13] J. A. Healey and R. W. Picard, "Detecting Stress During Real-World Driving Tasks Using Physiological Sensors," *IEEE Trans. Intell. Transp. Syst.*, vol. 6, no. 2, pp. 156–166, Jun. 2005, doi: 10.1109/TITS.2005.848368.

[14] N. Jaques et al, "Predicting students' happiness from physiology, phone, mobility, and behavioral data," in *2015 International Conference on Affective Computing and Intelligent Interaction (ACII)*, Xi'an, China, Sep. 2015, pp. 222–228. doi: 10.1109/ACII.2015.7344575.

[15] K. Yano *et al.*, "Profiting From IoT: The Key Is Very-Large-Scale Happiness Integration," *Symp. VLSI Circuits Dig. Tech. Pap.*, p. 4, 2015.

[16] A. Sano, P. Johns, and M. Czerwinski, "Designing Opportune Stress Intervention Delivery Timing using Multi-modal Data," *Int. Conf. Affect. Comput. Intell. Interact. ACII*, 2017.

[17] Y. Li, A. S. Elmaghraby, A. El-Baz, and E. M. Sokhadze, "Using physiological signal analysis to design affective VR games," in *2015 IEEE International Symposium on Signal Processing and Information Technology (ISSPIT)*, Abu Dhabi, United Arab Emirates, Dec. 2015, pp. 57–62.

[18] A. Jimenez-Molina *et al.*, "Using Psychophysiological Sensors to Assess Mental Workload During Web Browsing," *Sensors*, vol. 18, no. 2, p. 458, Feb. 2018.

[19] J. Wache, "The Secret Language of Our Body: Affect and Personality Recognition Using Physiological Signals," in *Proceedings of the 16th International Conference on Multimodal Interaction*, Istanbul Turkey, Nov. 2014, pp. 389–393. doi: 10.1145/2663204.2666290.

[20] S. R. Livingstone and F. A. Russo, "The Ryerson Audio-Visual Database of Emotional Speech and Song (RAVDESS): A dynamic, multimodal set of facial and vocal expressions in North American English," *PLOS ONE*, vol. 13, no. 5, p. e0196391, May 2018, doi: 10.1371/journal.pone.0196391.

[21] G. Costantini, I. Iadarola, A. Paoloni, and M. Todisco, "EMOVO Corpus: an Italian Emotional Speech Database," p. 4.

[22] H. Cao, D. G. Cooper, M. K. Keutmann, R. C. Gur, A. Nenkova, and R. Verma, "CREMA-D: Crowd-Sourced Emotional Multimodal Actors Dataset," *IEEE Trans. Affect. Comput.*, vol. 5, no. 4, pp. 377–390, Oct. 2014.

[23] K. Xu and C. S. Choy, "Low-power H.264/AVC baseline decoder for portable applications," in *Proceedings of the 2007 international symposium on Low power electronics and design - ISLPED '07*, Portland, OR, USA, 2007, pp. 256–261. doi: 10.1145/1283780.1283835.

[24] D. Hazer-Rau *et al.*, "The uulmMAC Database—A Multimodal Affective Corpus for Affective Computing in Human-Computer Interaction," *Sensors*, vol. 20, no. 8, p. 2308, Apr. 2020, doi: 10.3390/s20082308.

[25] C. Stachl *et al.*, "Predicting personality from patterns of behavior collected with smartphones," *Proc. Natl. Acad. Sci.*, vol. 117, no. 30, pp. 17680–17687, Jul. 2020, doi: 10.1073/pnas.1920484117.