From Pixels to Affect: A Study on Games and Player Experience

Konstantinos Makantasis
Institute of Digital Games
University of Malta,
konstantinos.makantasis@um.edu.mt

Antonios Liapis
Institute of Digital Games
University of Malta,
antonios.liapis@um.edu.mt

Georgios N. Yannakakis
Institute of Digital Games
University of Malta,
georgios.yannakakis@um.edu.mt

***Abstract*—Is it possible to predict the affect of a user just by observing her behavioral interaction through a video? How can we, for instance, predict a user's arousal in games by merely looking at the screen during play? In this paper we address these questions by employing three dissimilar deep convolutional neural network architectures in our attempt to learn the underlying mapping between video streams of gameplay and the player's arousal. We test the algorithms in an annotated dataset of 50 gameplay videos of a survival shooter game and evaluate the deep learned models' capacity to classify high vs low arousal levels. Our key findings with the demanding leave-one-video-out validation method reveal accuracies of over 78% on average and 98% at best. While this study focuses on games and player experience as a test domain, the findings and methodology are directly relevant to any affective computing area, introducing a general and user-agnostic approach for modeling affect.**

***Index Terms*—computer vision, gameplay footage, deep learning, arousal, affect classification**

I. Introduction

Designing general methods that are capable of performing equally well across various tasks has been a traditional vision of artificial intelligence [1]. A milestone study in that direction is the work of Mnih *et al.* [2] who achieved superhuman performance when playing several 2D games by merely observing the pixels of the screen. As impressive as these results might be, they are still limited to a particular set of tasks an agent needs to perform (i.e. play 2D Atari games) with clearly-defined objectives (i.e. maximize score). To which degree, however, could such general pixel-based representations learn to predict subjectively-defined notions such as emotion?

In this paper we attempt to address the above question based on the assumption that the behavior captured via the video of an interaction interweaves aspects of user experience that computer vision algorithms may detect. Thus, our key hypothesis is that we can construct accurate models of affect based only on the pixels of the interaction. In the current study we test this hypothesis in the domain of games by assuming that there is an unknown underlying function between what a player sees on the screen during a gameplay session and the level of arousal in the game. We use games as our initial domain in this endeavor, as gameplay videos have the unique property of overlaying the game context onto aspects of playing behavior and affect. Given that player affect is already embedded in the context of playing, the dominant affective computing practice suggesting the fusion of context with affect is not necessary in this domain [3]–[8]. Our approach is general and applicable to a variety of interaction domains beyond games since it only relies on decontextualized input (i.e. raw pixel values).

Given the spatio-temporal nature of the task, we use three types of deep convolutional neural network (CNN) architectures to classify between low and high values of annotated arousal traces based on a video frame or a video sequence. In particular, we test the CNNs in a dataset of 50 gameplay videos of a 3D survival shooter game. All videos have been annotated for arousal by the players themselves (first-person annotation) using the *RankTrace* [9] continuous annotation tool. Our key findings suggest that the task of predicting affect from the pixels of the experienced content is not only possible but also very accurate. Specifically, the obtained models of arousal are able to achieve average accuracies of over 78% using the demanding leave-one-video-out cross-validation method; the best models we obtained yield accuracies higher than 98%. The results also demonstrate—at least for the examined game—that player experience can be captured solely through on-screen pixels in a highly accurate and general fashion.

This paper is novel in several ways. First, this is the first attempt to model player affect just by observing the context of the interaction and not through any other direct manifestation of emotion or modality of user input; in that regard the solution we offer is *general* and *user-agnostic*. Second, to the best of our knowledge, this is the first time a study attempts to map directly from gameplay screen to game experience and infer a function between the two. Finally, three CNNs variants are compared for their ability to infer such a mapping in affective computing; the high accuracy values obtained demonstrate their suitability for the task.

II. Related Work

This section covers the related areas of affect modeling via videos, deep learning for images and videos, and affect modeling in games.

A. Video-Based Affect Modeling

Videos have been at the core of interest for both eliciting and modeling emotions in affective computing [10]. Typically, the video features a human face (or a group of faces) and emotion

This paper is funded, in part, by the H2020 project Com N Play Science (project no: 787476).

978-1-7281-3888-6/19/$31.00 ©2019 IEEE

is modelled through the detection of facial cues (see [3], [11], [12] among many) due to theoretical frameworks and evidence supporting that facial expressions can convey emotion [13]–[15]. Beyond the facial expression of a subject, aspects such as the body posture [16], [17], gestures [18] or gait [19], [20], have been used as input for modeling affect.

To estimate the affective responses elicited to a person by external stimuli, affect annotations of such responses are required naturally. Indicatively, Chen *et al.* [21] created a database of GIF animations, which users could rank across several affective dimensions, and modelled affect based on visual and tag features of the GIFs. In general, the onerous task of annotation makes such tasks "intrinsically a small-sample learning problem" [22]. This makes data-intensive methods such as deep learning rather inappropriate. However, recent advances in deep learning have spurred research interest in emotion expression corpora, with several medium- and large-scale datasets as surveyed in [23]. CNNs were first applied in [24] to predict dimensional affective scores from videos, but the issue of small samples (raised above) challenged CNN learning. In [25], CNNs were combined with recurrent neural networks to model arousal-valence using the *Aff-Wild* database [26]. In [27] the authors exploit deep end-to-end trainable networks for recognizing affect in real-world environments. Finally, McDuff *et al.* [3] fused facial expression data and videos of advertisements to classify whether viewers liked the videos or were willing to view them again.

The modeling work presented here is unconventional within the broader affective computing field as it utilizes videos as both the elicitor of emotion and the sole modality for modeling affect. In a sense, what we achieve with the proposed approach is a general method for modeling affect via videos, as neither facial nor bodily expression is available as input to the affect model. The obtained high accuracies—at least within the games domain— suggest that this subject-agnostic perspective is not only possible but it also yields models of high predictive capacity.

B. Deep Learning for Images and Videos

Conventional machine learning methods have often been used for pattern recognition in images, videos and other data types, but have been held back by the requirement that raw data needed to be transformed to a suitable representation via a handcrafted feature construction process based on expert knowledge. The recent success of *deep learning* [28] approaches is largely due to their ability to learn representations directly from the raw data via the composition of simple but nonlinear data transformations. Very complex functions can be learned by combining enough transformations, and deep learning has shown tremendous success in visual recognition [29], natural language processing [30] and agent control [2].

Convolutional neural networks are deep learning models which apply two-dimensional trainable filters and pooling operations on the raw input, resulting in a hierarchy of increasingly complex features. By design, CNNs are able to encode the spatial information of their inputs. CNNs are therefore particularly powerful in discovering patterns in 2D images [29]. CNNs have also been applied for classification of video sequences, similar to this paper, using a frame-by-frame input or a 3D representation with a temporal dimension [31], [32]. While Jia *et al.* [33] first used 3D CNNs on cropped parts of a video, Ji *et al.* proposed 3D CNNs for any video classification under the assumption that "2D ConvNets lose temporal information of the input signal right after every convolution operation" [32]. The testbed of Ji *et al.* was the C3D dataset ($1.1 \cdot 10^6$ videos of 487 sports categories) with video frames resized to 128×171 pixels. The seminal paper of Karpathy *et al.* [31] explored several architectures for fusing information over the temporal dimension, including an early fusion approach which combined RGB channels over time (as 4D CNN) and a late fusion which used two single-frame networks (each receiving frames spaced half a second apart) and compared outputs to derive global motion characteristics. Similarly to [32], the work of Karpathy *et al.* was also tested on the C3D dataset, but videos were resized and cropped to 170×170 pixels.

This paper uses a game footage dataset which is far smaller than the data available to the above studies, which necessitated a simplification of both the video input (which was downscaled more aggressively and only used the brightness channel) and the CNN architecture (with far fewer trainable parameters).

C. Affect Modeling in Games

Player modeling is the study of computational models of players, their behavioral patterns and affective responses [34]. If target outputs are available, a player model considers some input modality regarding the player (e.g. their gameplay and physiology) and is trained to predict aspects of the in-game behavior or the player experience. Indicatively, in studies with *Super Mario Bros.* (Nintendo, 1985) gameplay data (e.g. number of deaths) combined with level features (e.g. number of gaps) [35], or the player's posture during gameplay [36] were used to predict the player's reported affect.

This study advances the state of the art in player modelling by using solely raw gameplay information to model a player's emotions. Within the broader area of artificial intelligence and games [37], the majority of the works that analyse and extract information from gameplay videos focus on inferring the strategy, structure and the physics of the games themselves [2], [38]. In this work, instead, we use the same kind of information for modelling a player's experience in a general fashion (from pixels to experience), ignoring the game per se. At the same time, the most common approaches for analysing player experience, besides game and gameplay information, heavily rely on direct measurements from players, such as face monitoring, speech and physiological signals; see e.g. [4], [36], [39]). Unlike these approaches, our methodology relies solely of gameplay video information. This critical difference advances player experience modelling as the approach does not require access to intrusive player measurements collected under well-defined experimental settings, thus allowing the vast collection of data. As gameplay videos are already avail-

able over the web and produced daily in massive amounts, the approach is feasible and can potentially generalize to any game.

III. Methodology

This paper explores the degree to which frames and videos of gameplay footage can act as the sole predictors of a player's affective state. This section describes the gameplay dataset and how it was collected, the employed CNN architectures, as well as the dataset preparation process for training the CNNs.

A. Dataset Description

The gameplay videos we used in the experiments of this paper are captured from a shooter game developed in the *Unity 3D* game engine. Specifically, we use the **Survival Shooter** [40], which is a game adapted from a tutorial package of Unity 3D. In this game the player has 60 seconds to shoot down as many hostile toys as possible and avoid running out of health due to toys colliding with the avatar. Hostile toys keep spawning at predetermined areas of the level and converge towards the player. The player's avatar has a gun that shoots bright laser beams, and can kill each toy with a few shots. Every toy killed adds to the player's score.

The data was collected from 25 different players who each produced and annotated two gameplay videos. Each player played a game session (60 seconds) and then annotated their recorded gameplay footage in terms of arousal. Annotation was carried out using the *RankTrace* annotation tool [9] which allows the continuous and unbounded annotation of affect using the Griffin PowerMate wheel interface. Gameplay videos were captured at 30Hz (i.e. 30 frames per second) while the RankTrace tool provided four annotation samples per second. Figure 1 shows three indicative frames of the Survival Shooter gameplay and the annotations of arousal from RankTrace.

The corpus of gameplay videos was cleaned by omitting gameplay footage under 15 seconds, resulting in a clean corpus of 45 gameplay videos and a total of $8,093$ annotations of arousal. While the average duration of playthroughs in this corpus is 44 seconds, in 60% of the playthroughs the player survived for the full 60 seconds and completed the game level.

B. Training Data Preparation

In order to evaluate how CNNs can map raw video data to affective states, we train CNN models using as input individual frames that contain only spatial information, and video segments that contain both spatial and temporal information. This section describes the input and the output of the networks.

Since RankTrace provides unbounded annotations, we first convert the annotation values of each video to $[0, 1]$ via min-max normalization and synchronize the recording frequency of videos (30Hz) with annotations (4Hz) by treating the arousal value of any frame without an annotation as the arousal value of the last annotated frame. In order to decrease the computational complexity of training and evaluating CNNs, we convert RGB video frames to grayscale and resize them to 72×128 pixels; this results in a more compact representation which considers only the brightness of the image and not its color. Due to the stark shadows and brightly lit avatar and projectiles in the Survivor Shooter, we consider that brightness is likely a core feature for extracting gameplay behavior. While RGB channels or a larger frame size could provide more information about the gameplay and affect dimensions, it would require substantially more data for CNNs to train on.

Regarding the input of the CNN, we have to decide which frames and video segments will be used as input points. Schindler and Van Gool [41] argue that a small number of subsequent frames are adequate to capture the content of a scene. Based on this argument, the authors of [42] achieve high human activity recognition rates by describing an activity with mini video batches of 8 subsequent frames. Motivated by these works, we also use 8 subsequent frames to characterize the player's state of affect. Specifically, the gameplay videos are split into non-overlapping segments of 8 subsequent frames which are used as input to the temporally aware CNN architectures. If the input is a single image, the last frame of each video segment is used.

The output of the CNN is straightforward to compute based on the 8-frame video segments. Since annotations are made at 4Hz, in most cases a video frame segment would include one annotation. In cases where two annotations are given within 8 frames, their average value is computed. RankTrace produces interval data and thus it may seem natural to state the problem as a regression task; given that we aim to offer a user-agnostic and general approach, however, we do not wish to make any assumptions regarding the value of the output as this may result in highly biased and user-specific models [43]. For this reason we state our problem as a classification task and transform interval values into binary classes (low and high arousal) by using the mean value of each trace as the class splitting criterion (see Fig. 1). The class split may use an optional threshold parameter (ϵ) to determine the zone within which arousal values around the mean are labelled as 'uncertain' and ignored during classification. Detailed experiments with the ϵ parameter are conducted in Section IV-B. While alternative ways of splitting the classes were considered (such as the area under the curve or the median), in this paper we include only experiments with the most intuitive way to split such a trace given its unbounded nature: its mean.

C. CNN Architectures

In this study we explore three different CNN architectures. The first two apply 2D trainable filters on the inputs (single frames or videos), while the third applies 3D trainable filters. All CNN architectures have the same number of convolutional and fully connected layers, the same number of filters at their corresponding convolutional layers and the same number of hidden neurons at their fully connected layer. This way we are able to fairly compare the skill of the three architectures to map video data to affective states, and at the same time to gain insights on the effect of temporal information to the classification task. It should be noted that current state-of-the-art CNNs for videos and images alike use much larger

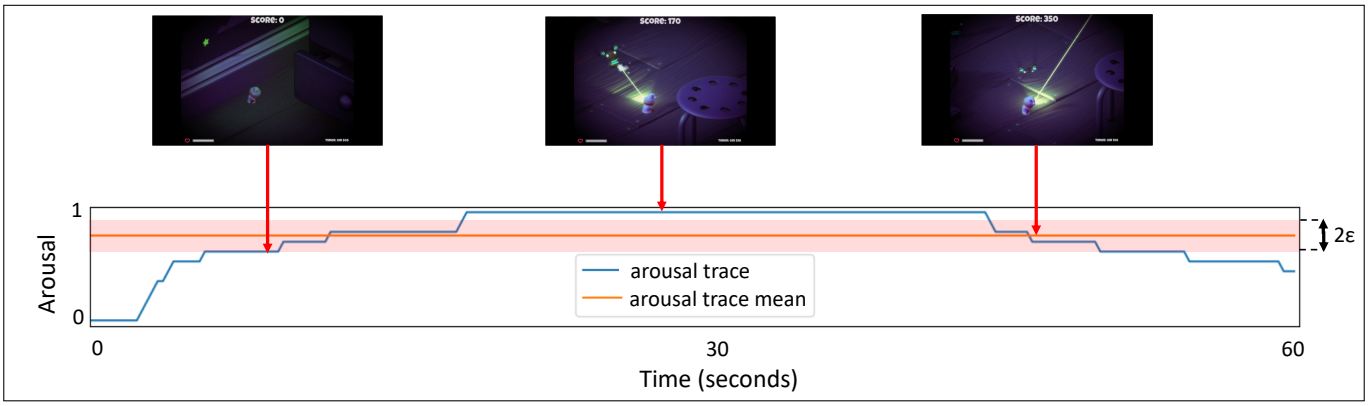


Fig. 1. The normalized to $[0, 1]$ trace of affect (arousal) produced by RankTrace, the uncertainty zone define by ϵ, and three indicative frames of one of the Survival Shooter gameplays.

architectures (e.g. [31]); in this paper, however, we explore more compact architectures due to the small size of the dataset.

1) 2DFrameCNN: The first CNN architecture (see Fig. 2) uses as input a single frame on which it applies 2D filters. The *2DFrameCNN* architecture consists of three convolutional layers with 8, 12 and 16 filters, respectively, of size 5×5 pixels. Each convolutional layers is followed by a 2D max pooling layer of size 2×2. The output of convolutions is a feature vector of 960 elements, which is fed to a fully connected layer with 64 hidden neurons that connect to the output. This architecture has approximately $6.9 \cdot 10^4$ trainable parameters and exploits only the spatial information of the video data.

2) 2DSeqCNN: The second CNN architecture applies 2D filters to input video segments. The *2DSeqCNN* network has exactly the same topology as the 2DFrameCNN architecture but the number of trainable parameters is slightly higher (approximately $7 \cdot 10^4$) as the inputs are video sequences. This architecture implicitly exploits both the spatial and the temporal information of the data.

3) 3DSeqCNN: The third CNN architecture applies 3D filters to input video segments. As with the other architectures, *3DSeqCNN* has three convolutional layers with 8, 12 and 16 filters, respectively, of size $5 \times 5 \times 2$ pixels. Each one of the convolutional layers is followed by a 3D max pooling layer of size $2 \times 2 \times 1$. The 3D convolutional layers produce a feature vector of 1,920 elements, which is fed to a fully connected layer with 64 neurons. Due to its 3D trainable filters, *3DSeqCNN* has approximately $14.5 \cdot 10^4$ trainable parameters. This architecture explicitly exploits both the spatial and the temporal information of the data due to the application of the trainable filter along the spatial and the temporal dimensions.

While 2DFrameCNN receives as input a single frame, both 2DSeqCNN and 3DSeqCNN receive as input a sequence of 8 frames, i.e. a time slice of the video lasting 267 milliseconds. In all three network architectures, we apply batch normalization on the features constructed by the convolutional layers before feeding them to the last fully connected layer, which in turn feeds two output neurons for binary classification. All of the hyperparameters of the CNN architectures are manually selected in an attempt to balance two different criteria: (a) computational complexity (training and evaluation times), and (b) learning complexity (ability to avoid under-/over-fitting).

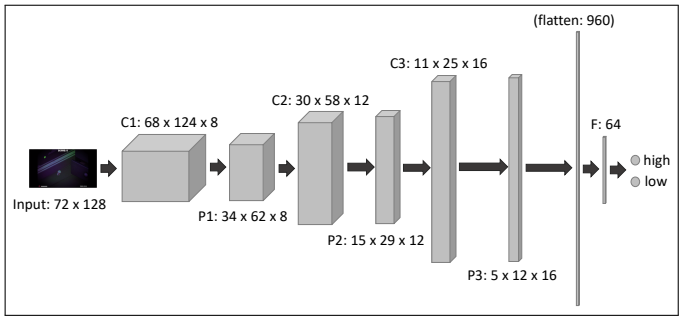


Fig. 2. The architecture of 2DFrameCNN. Convolutional layers are denoted with a "C", max pooling layers with a "P", and fully connected layers with an "F".

IV. Experiments

To test our hypothesis that there is a learnable underlying function between affect and its visual manifestations on gameplay videos, in this section we use the three CNNs for classifying gameplay footage as *high* or *low* arousal (as discussed in Section III-B). As mentioned earlier, this binary classification approach is well-suited for unbounded and continuous traces (as the mean of each annotation trace is different), and can produce a sufficiently rich dataset for deep learning. Section IV-A explores the performance of different CNN architectures on this naive split between high and low arousal, while Section IV-B explores the impact of an uncertainty bound that filters out segments that are too close to the mean arousal value.

In all reported experiments, we follow the demanding *leave-one-video-out* scheme [3]; this means that we use data from 44 videos to train the models and then we evaluate their performance on the data from the video that is not used for training (i.e. test set). This procedure is repeated 45 times until we test the performance of CNNs on the data from all videos. During the training of the models we also employ early stopping criteria to avoid overfitting. For early stopping, data of the 44 videos is shuffled and split further into a training set (90% of the data) and a validation set for testing overfitting

TABLE I
TEST ACCURACY FOR BINARY CLASSIFICATION OF DIFFERENT CNN ARCHITECTURES, AND FOR DIFFERENT THRESHOLD VALUES FOR CLASSIFICATION (ϵ). THE 95% CONFIDENCE INTERVAL IS INCLUDED.

ϵ	Baseline	2DFrameCNN	2DSeqCNN	3DSeqCNN
0.00	51%±0.0%	70%±4.2%	74%±4.7%	73%±4.4%
0.05	56%±0.3%	72%±5.6%	73%±5%	73%±5.3%
0.10	55%±0.3%	74%±5.7%	75%±5.6%	74%±5.7%
0.20	50%±0.3%	77%±5.7%	78%±5.6%	77%±5.7%

(10% of the data). Early stopping is activated if the loss on the validation set does not improve for 15 training epochs. Reported accuracy is the classification accuracy on the test set, averaged from 45 runs. Significance is derived from the 95% confidence interval of this test accuracy. The baseline accuracy is the average classification accuracy on the test set, when we always select the most common class in the 44 videos of the training set. Naturally, the baseline also indicates the distribution of the ground truth between the two classes.

A. Binary Classification of Arousal

The most straightforward way to classify segments of gameplay footage is based on the mean arousal value of the annotation trace, treating all annotations above the mean value as high arousal and below it as low arousal. This naive classification results to a total of $8,093$ data points (i.e. 8-frame segments assigned to a class) from all 45 videos.

The top row of Table I reports the average classification accuracy of the CNN models with the naive classification method ($\epsilon = 0$). All models have accuracies over 20% higher than the baseline classifier, which suggests that CNNs, regardless of the architecture used, have the capacity to map raw gameplay video to arousal binary states. The model that performs best is the 2DSeqCNN, which implicitly exploits the temporal information in the data. Its accuracy is over 3% higher than the 2DFrameCNN which exploits only spatial information, but it is only slightly better than the 3DSeqCNN. The ability of the 3DSeqCNN to explicitly exploit the temporal information does not seem to significantly affect its performance. Comparing the performance of the 2DFrameCNN with the performances of the other two CNN models indicates that although the temporal information contributes to the learning process, the dominant information of the inputs comes from their spatial and not their temporal structure. This may be due to the very short duration of the input video segments (267 milliseconds), or due to strong predictors of arousal existent in the heads-up display of the game (see Section IV-C).

B. Exploring the Uncertainty Bound of Arousal

While classifying all data above the mean value of the arousal trace as high yields a large dataset, the somewhat arbitrary split of the dataset may misrepresent the underlying ground truth and also introduce split criterion biases [43], [44]. Specifically, frames with arousal values around the mean would be classified as *high* or *low* based on trivial differences. To filter out annotations that are ambiguous (i.e. close to the man arousal value $\hat{A}$), we use the ϵ value and omit any datapoints with an arousal value A within the *uncertainty bound* determined by ϵ: $\hat{A} - \epsilon < A < \hat{A} + \epsilon$ (see Fig. 1 for a graphical depiction). This section tests how the performance of the three CNN classifiers changes when three different threshold values $\epsilon = \{0.05, 0.10, 0.20\}$ remove ambiguous data points from the dataset.

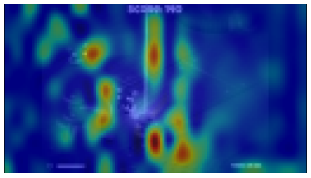
Table I shows the performance of different CNN architectures for different threshold values. It should be noted that removing datapoints affects the baseline values quite substantially as representatives of one class become more frequent than for the other class. Regardless, we see that the accuracy of all architectures increases when data with ambiguous arousal values is removed, especially for higher ϵ values. For $\epsilon = 0.20$, the accuracy of all three CNN architectures is 26% to 28% higher than the baseline. The 2DFrameCNN also benefits from the cleaner dataset, being second in accuracy only to 2DSeqCNN for $\epsilon = 0.10$ and $\epsilon = 0.20$. The additional trainable parameters of 3DSeqCNN seem to require more data than what is available in the sparser datasets. Indeed, the number of total datapoints decreases by 12% for $\epsilon = 0.05$, by 25% for $\epsilon = 0.10$, and by 44% for $\epsilon = 0.20$ (for a total of $4,534$ datapoints). It is obvious that having a cleaner but more compact dataset can allow the less complex architectures (2DFrameCNN, 2DSeqCNN) to derive more accurate models but can challenge complex architectures (3DSeqCNN). The trade-off poses an interesting problem moving forward for similar tasks of gameplay annotation.

C. Analysis of Findings

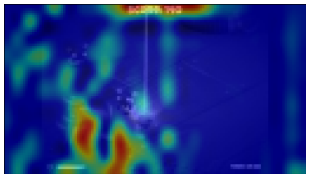
Experiments showed that it is possible to produce surprisingly accurate models of players' arousal from on-screen gameplay footage alone—even from a single frame snapshot. Especially when removing data with ambiguous arousal annotations, a model of 2DFrameCNN can reach a test accuracy of 98% (at $\epsilon = 0.20$), although on average the test accuracy is at 77%. It is more interesting, however, to observe which features of the screen differentiate frames or videos into low-arousal or high-arousal classes. This can be achieved by showing which parts of the frame have the most influence on the model's prediction, e.g. via Gradient-weighted Class Activation Mapping [45]. This method computes the gradient of an output node with respect to the nodes of a convolutional layer, given a particular input. By multiplying the input with the gradient, averaging over all nodes in the layer and normalizing the resulting values, we obtain a heatmap that shows how much each area of the input contributed to increasing the value of the output node.

Figure 3 shows the activation maps for low versus high arousal of a sample gameplay frame, calculated based on the 2DFrameCNN. While 2DSeqCNN has higher accuracies, it is far more challenging to visually capture the sequence on paper so we opt for the frame-only information of 2DFrameCNN. We immediately observe that both low and high arousal predictors focus on aspects of the heads-up display (HUD) which are overlaid on the 3D world where the player navigates,

(a) Frame of gameplay footage (in full resolution and full-color)



(b) Activation of **Low** Arousal



(c) Activation of **High** Arousal

Fig. 3. Activation maps for a sample frame of the game

shoots and collides with hostile toys. Specifically, the score at the top center of the screen contributes substantially to high arousal. Interestingly, the score keeps increasing during the progression of the game as the player kills more and more hostile toys. The impact of time passed in the game—and by extent increasing score—on arousal can be corroborated by the annotations themselves: in most cases the annotators kept increasing the arousal level as time went by rather than decreasing it. Tellingly, of all arousal value changes in the entire dataset, 807 instances were increases and 297 were decreases. Thus, both score and time remaining would be simple indicators of low or high arousal. Interestingly, the HUD element of the player's health was not considered for either class. Among other features of the 3D gameworld, hostile toys are captured by the low arousal output, while an obstacle next to the player is captured by the high arousal output. It is less clear what other areas activated on the screens of Fig. 3 capture with regards to arousal.

V. Discussion

This paper presented the first attempt, to our knowledge, of modelling affect solely via videos that do not display human behavior *directly*; such videos of interaction instead display human behavior in an *indirect* manner as emotion is manifested through and annotated on the video per se. We also introduced the first modeling attempt of players' affective states based on the on-screen captured gameplay alone. Using a time window of 8 frames and selecting either a single frame or the frame sequence within that time window, a number of CNN architectures were tested. Results show that processing the gameplay footage as short videos results in higher classification accuracy, although in general all three models perform comparably. Moreover, when data within an uncertainty bound around the trace's mean arousal are not considered, the smaller remaining dataset challenges 3D convolutional layers but yields highly accurate models (approximately 78% accuracy, on average) for simpler networks based on frames or frame-by-frame processing of videos. Despite this paper being a first attempt at a challenging task of predicting player affect from gameplay pixels, the results are promising and point to a number of extensions in future work. We discuss these below.

As this was an initial exploratory study, there is a number of assumptions made for both the input and the output of the affect model. In terms of input, we used only the brightness channel of the gameplay footage; in part, this was because of the structure of the game itself, due to the high-contrast "horror" aesthetic, and because one channel allowed us to train models faster and with the few data points at hand. Future experiments, however, should explore other formats for CNN-based video classification in the literature, such as scaling the input to be a higher-resolution image, and using hand-crafted channels that include edge detection [32] or RGB channels [31]. While the goal of this study was to detect player arousal from gameplay footage alone, future studies could explore how fusing gameplay footage with other information streams such as gameplay logs and physiological data [40] would affect the model's accuracy. In terms of the affect labels (output), taking the mean arousal value (normalized to a player's full trace) within a time window was an intuitive solution, but could be expanded on and refined. More relative ways of processing annotations within a time window, such as amplitude and average gradient [9], [40], could be explored. Moreover, the video information (processed through a CNN) could be used to predict not the class of high or low arousal but instead whether there is an increase or decrease from the previous time window. This method would better align with the temporal nature of gameplay videos, but would likely decrease the size of the dataset as many subsequent time windows have the same mean arousal value (see Fig. 1). Finally, using a sliding time window rather than a non-overlapping window of 8 frames would increase the size of the dataset and perhaps better capture all annotations.

VI. Conclusion

In this paper we introduced a general method that captures affect solely from videos which embed forms of human computer interaction but without humans explicitly depicted in the video. Using games as our domain, we explored how gameplay footage can be processed and fed to three different convolutional neural network architectures that, in turn, predict a player's arousal levels in a binary fashion. The obtained models of arousal trained this way yield accuracies of up to 78% on average (98% at best). Our analysis also reveals the different on-screen aspects that contribute to higher vs. lower arousal in the testbed game. While this initial study focuses on games as a domain, the findings and methodology are directly relevant to any affective computing area, introducing a general and user-agnostic approach for modeling affect.

REFERENCES

[1] Ben Goertzel and Cassio Pennachin, *Artificial general intelligence*, vol. 2, Springer, 2007.

[2] Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Andrei A Rusu, Joel Veness, Marc G Bellemare, Alex Graves, Martin Riedmiller, Andreas K Fidjeland, Georg Ostrovski, et al., "Human-level control through deep reinforcement learning," *Nature*, vol. 518, no. 7540, 2015.

[3] Daniel McDuff, Rana el Kaliouby, David Demirdjian, and Rosalind Picard, "Predicting online media effectiveness based on smile responsesgathered over the internet," in *Image and Vision Computing*. Elsevier, 2014.

[4] Charles Ringer and Mihalis A Nicolaou, "Deep unsupervised multi-view detection of video game stream highlights," in *Proceedings of FDG*, 2018.

[5] Zhihong Zeng, Maja Pantic, Glenn I Roisman, and Thomas S Huang, "A survey of affect recognition methods: Audio, visual, and spontaneous expressions," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 31, no. 1, pp. 39–58, 2009.

[6] Daniel McDuff, Rana El Kaliouby, Karim Kassam, and Rosalind Picard, "Affect valence inference from facial action unit spectrograms," in *Proc. of the IEEE Computer Society Conf. on Computer Vision and Pattern Recognition Workshops*. IEEE, 2010, pp. 17–24.

[7] Lazaros Zafeiriou, Stefanos Zafeiriou, and Maja Pantic, "Deep analysis of facial behavioral dynamics," in *Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition Workshops*, 2017.

[8] Martin Wöllmer, Felix Weninger, Tobias Knaup, Björn Schuller, Congkai Sun, Kenji Sagae, and Louis-Philippe Morency, "Youtube movie reviews: Sentiment analysis in an audio-visual context," *IEEE Intelligent Systems*, vol. 28, no. 3, pp. 46–53, 2013.

[9] Phil Lopes, Georgios N Yannakakis, and Antonios Liapis, "Ranktrace: Relative and unbounded affect annotation," in *Proc. of the Intl. Conf. on Affective Computing and Intelligent Interaction*, 2017, pp. 158–163.

[10] Rosalind Picard, "Affective computing," Tech. Rep., MIT, 1995.

[11] G. Littlewort, J. Whitehill, T. Wu, I. Fasel, M. Frank, J. Movellan, and M. Bartlett, "The computer expression recognition toolbox (CERT)," in *Proc. of Face and Gesture*, 2011.

[12] M. Bartlett, G. Littlewort, M. Frank, C. Lainscsek, I. Fasel, and J. Movellan, "Automatic recognition of facial actions in spontaneous expressions," *Journal of Multimedia*, vol. 1, no. 6, 2006.

[13] Z. Ambadar, J. Schooler, and J. Cohn, "Deciphering the enigmatic face. the importance of facial dynamics in interpreting subtle facial expressions," *Psychological Science*, vol. 16, 2005.

[14] J. Bassili, "Emotion recognition: The role of facial movement and the relative importance of upper and lower areas of the face," *Journal of personality and social psychology*, vol. 37, no. 11, 1979.

[15] Paul Ekman, "An argument for basic emotions," *Cognition & emotion*, vol. 6, no. 3-4, pp. 169–200, 1992.

[16] Andrea Kleinsmith and Nadia Bianchi-Berthouze, "Recognizing affective dimensions from body posture," in *Proc. of the Intl. Conf. on Affective Computing and Intelligent Interaction*, 2007.

[17] Bianchi-berthouze N. Steed A. Kleinsmith, A., "Automatic recognition of non-acted affective postures," *IEEE Trans. on Systems, Man and Cybernetics*, 2011.

[18] D. Glowinski, A. Camurri, G. Volpe, N. Dael, and K. Scherer, "Technique for automatic emotion recognition by body gesture analysis," in *Proc. of the IEEE Computer Society Conf. on Computer Vision and Pattern Recognition Workshops*, 2008.

[19] Joann M. Montepare, Sabra B. Goldstein, and Annmarie Clausen, "The identification of emotions from gait information," *Journal of Nonverbal Behavior*, vol. 11, no. 1, pp. 33–42, 1987.

[20] Shun Li, Liqing Cui, Changye Zhu, Baobin Li, Nan Zhao, and Tingshao Zhu, "Emotion recognition using Kinect motion capture data of human gaits," *PeerJ*, 2016.

[21] W. Chen, O. O. Rudovic, and R. W. Picard, "GIFGIF+: collecting emotional animated GIFs with clustered multi-task learning," in *Proc. of ACII*, 2017, pp. 510–517.

[22] N. Li, Y. Xia, and Y. Xia, "Semi-supervised emotional classification of color images by learning from cloud," in *Proc. of the Intl. Conf. on Affective Computing and Intelligent Interaction*, 2015, pp. 84–90.

[23] P. Barros, N. Churamani, E. Lakomkin, H. Siqueira, A. Sutherland, and S. Wermter, "The OMG-Emotion behavior dataset," in *Proc. of the Intl. Joint Conf. on Neural Networks*, 2018.

[24] Y. Baveye, E. Dellandrea, C. Chamaret, and L. Chen, "Deep learning vs. kernel methods: Performance for emotion prediction in videos," in *Proc. of ACII*, 2015, pp. 77–83.

[25] Dimitrios Kollias, Mihalis A Nicolaou, Irene Kotsia, Guoying Zhao, and Stefanos Zafeiriou, "Recognition of affect in the wild using deep neural networks," in *Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition Workshops*, 2017, pp. 26–33.

[26] Stefanos Zafeiriou, Dimitrios Kollias, Mihalis A Nicolaou, Athanasios Papaioannou, Guoying Zhao, and Irene Kotsia, "Aff-wild: Valence and arousal 'in-the-wild' challenge," in *Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition Workshops*, 2017, pp. 34–41.

[27] Panagiotis Tzirakis, Stefanos Zafeiriou, and Björn Schuller, "Real-world automatic continuous affect recognition from audiovisual signals," in *Multimodal Behavior Analysis in the Wild*. Elsevier, 2019.

[28] Ian Goodfellow, Yoshua Bengio, and Aaron Courville, *Deep learning*, MIT press, 2016.

[29] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in neural information processing systems*, 2012, pp. 1097–1105.

[30] Tom Young, Devamanyu Hazarika, Soujanya Poria, and Erik Cambria, "Recent trends in deep learning based natural language processing," *IEEE Computational Intelligence Magazine*, vol. 13, no. 3, 2018.

[31] Andrej Karpathy, George Toderici, Sanketh Shetty, Thomas Leung, Rahul Sukthankar, and Li Fei-Fei, "Large-scale video classification with convolutional neural networks," in *Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition*, 2014.

[32] Shuiwang Ji, Wei Xu, Ming Yang, and Kai Yu, "3d convolutional neural networks for human action recognition," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 35, pp. 221231, 2013.

[33] Yangqing Jia, Evan Shelhamer, Jeff Donahue, Sergey Karayev, Jonathan Long, Ross Girshick, Sergio Guadarrama, and Trevor Darrell, "Caffe: Convolutional Architecture for Fast Feature Embedding," *arXiv e-prints*, Jun 2014.

[34] Georgios N. Yannakakis, Pieter Spronck, Daniele Loiacono, and Elisabeth André, "Player modeling," in *Artificial and Computational Intelligence in Games (Dagstuhl Seminar 12191)*, pp. 45–59. 2012.

[35] Chris Pedersen, Julian Togelius, and Georgios N. Yannakakis, "Modeling player experience in super mario bros," in *Proc. of the Intl. Conf. on Computational Intelligence and Games*, 2009.

[36] Noor Shaker, Stylianos Asteriadis, Georgios N. Yannakakis, and Kostas Karpouzis, "Fusing visual and behavioral cues for modeling user experience in games," *IEEE Trans. on System, Man and Cybernetics*, vol. 43, no. 6, 2013.

[37] Georgios N. Yannakakis and Julian Togelius, *Artificial Intelligence and Games*, Springer, 2018, http://gameaibook.org.

[38] Matthew Guzdial, Nathan Sturtevant, and Boyang Li, "Deep static and dynamic level analysis: A study on infinite mario," in *Proc. of the AIIDE workshop on Experimental AI in Games*, 2016.

[39] Hector P Martinez, Yoshua Bengio, and Georgios N Yannakakis, "Learning deep physiological models of affect," *IEEE Computational Intelligence Magazine*, vol. 8, no. 2, pp. 20–33, 2013.

[40] Elizabeth Camilleri, Georgios N. Yannakakis, and Antonios Liapis, "Towards general models of player affect," in *Proc. of the Intl. Conf. on Affective Computing and Intelligent Interaction*, 2017.

[41] Konrad Schindler and Luc J Van Gool, "Action snippets: How many frames does human action recognition require?," in *Proc. of the IEEE Intl. Conf. on Computer Vision and Pattern Recognition*, 2008.

[42] Konstantinos Makantasis, Anastasios Doulamis, Nikolaos Doulamis, and Konstantinos Psychas, "Deep learning based human behavior recognition in industrial workflows," in *Proc. of the Intl. Conf. on Image Processing*. IEEE, 2016, pp. 1609–1613.

[43] Georgios N Yannakakis, Roddy Cowie, and Carlos Busso, "The Ordinal Nature of Emotions: An emerging approach," *IEEE Trans. on Affective Computing*, 2018.

[44] Hector P Martinez, Georgios N Yannakakis, and John Hallam, "Don't classify ratings of affect; rank them!," *IEEE Trans. on Affective Computing*, vol. 5, no. 3, pp. 314–326, 2014.

[45] Ramprasaath R Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra, "Grad-cam: Visual explanations from deep networks via gradient-based localization," in *Proc. of the IEEE Intl. Conf. on Computer Vision*, 2017.