IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received September 3, 2021, accepted September 18, 2021, date of publication September 20, 2021,
date of current version October 4, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3114389

Keeping Children Safe Online With Limited Resources: Analyzing What is Seen and Heard

ALEKSANDAR JEVREMOVIC[1], MLADEN VEINOVIC[1], MILAN CABARKAPA[2], MARKO KRSTIC[2], (Member, IEEE), IVAN CHORBEV[3], IVICA DIMITROVSKI[3], NUNO GARCIA[4], NUNO POMBO[4], (Senior Member, IEEE), AND MILOS STOJMENOVIC[1]

[1]Department of Computer Science and Electrical Engineering, Singidunum University, 11000 Belgrade, Serbia
[2]School of Electrical Engineering, University of Belgrade, 11120 Belgrade, Serbia
[3]Faculty of Computer Science and Engineering, Ss. Cyril and Methodius University in Skopje, 1000 Skopje, North Macedonia
[4]Department of Computer Science, University of Beira Interior, 6201-001 Covilhã, Portugal

Corresponding author: Milos Stojmenovic (mstojmenovic@singidunum.ac.rs)

This work was supported by European Union's Horizon 2020 Research and Innovation Program through Next Generation Internet (NGI) Trust under Grant 825618.

ABSTRACT It is every parent's wish to protect their children from online pornography, cyber bullying and cyber predators. Several existing approaches analyze a limited amount of information stemming from the interactions of the child with the corresponding online party. Some restrict access to websites based on a blacklist of known forbidden URLs, others attempt to parse and analyze the exchanged multimedia content between the two parties. However, new URLs can be used to circumvent a blacklist, and images, video, and text can individually appear to be safe, but need to be judged jointly. We propose a highly modular framework of analyzing content in its final form at the user interface, or Human Computer Interaction (HCI) layer, as it appears before the child: on the screen and through the speakers. Our approach is to produce Children's Agents for Secure and Privacy Enhanced Reaction (CASPER), which analyzes screen captures and audio signals in real time in order to make a decision based on all of the information at its disposal, with limited hardware capabilities. We employ a collection of deep learning techniques for image, audio and text processing in order to categorize visual content as pornographic or neutral, and textual content as cyberbullying or neutral. We additionally contribute a custom dataset that offers a wide spectrum of objectionable content for evaluation and training purposes. CASPER demonstrates an average accuracy of 88% and an F1 score of 0.85 when classifying text, and an accuracy of 95% when classifying pornography.

INDEX TERMS Cyber-bullying, cyber-grooming, online safety, pornography filter, real time agent.

I. INTRODUCTION

Keeping children safe online is of paramount importance when so many spend a majority of their time in front of a computer or digital device, especially now due to school closures, and other government mandates in western countries. Depending on the level of government restriction, children cannot meet with their friends nor family members, and are limited to communicating via Voice over IP (VoIP) platforms or social media. This type of online life exposes children to various types of potentially dangerous interactions with unknown parties, many of which may degrade their mental as well as physical safety in extreme cases. By keeping children safe online, we mean minimizing their exposure to, and giving them tools to defend against online threats such as cyber-bullying, pornography, cyber-grooming, and the induction to self harm, etc.

This is a widely studied field, both from the psychological standpoint of how these types of dangers affect children's well-being [35], as well as from a cyber-security perspective, which aims to detect and prevent the occurrence of unwanted content. Individual systems exist that detect unwanted imagery [19], video [41], text [17], and audio [25]. There are also advances in this field that focus on identifying the individual types of threats listed above. Mallmann *et al.* [28] propose a system that tries to comprehensively defend against all types of threats by analyzing incoming content, but relies on hardware that may not be ubiquitously present across all devices. There exist systems that restrict access to certain manually blacklisted websites,

The associate editor coordinating the review of this manuscript and approving it for publication was Xiaojie Su.

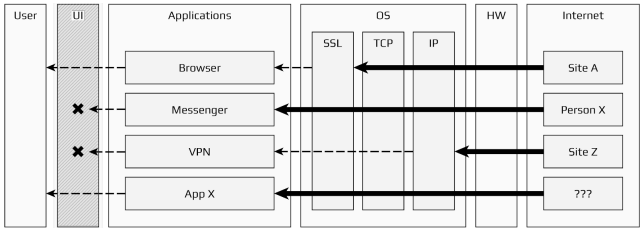


FIGURE 1. The UI layer of information between the user and the counter-party is captured and analyzed.

but such solutions are ineffective for obvious reasons. A system capable of directly looking at the content the user sees, and making decisions based on this final layer of information is the most comprehensive approach to solving this problem.

We propose a modular, lightweight system that takes all available User Interface (UI) outputs and jointly analyzes them to identify unwanted content for children. Our solution runs on hardware that relies solely on Central Processing Unit (CPU) processing, and does have access to a graphics card. This restriction makes our solution deployable on a wide range of platforms, increasing its ubiquity, while sacrificing some accuracy and performance. Our system is conceptually quite simple, as it categorizes content as it appears on the monitor and through the speakers in parallel. As seen in Fig 1, we only consider the analysis of information in the darkened UI layer, as it simplifies the problem, and allows us to 'see' exactly what the user sees.

We apply our algorithm on a set of captured browsing sessions which contain both appropriate and inappropriate material. We measure success by our ability to detect inappropriate parts of the manually labelled data-set. Our main contributions are our modular system architecture, as well as our manually labelled data-set, which was used for both training, validation and testing or our image detection algorithm from screenshots. This dataset can also be used for pornography detection as the manually annotated images also contain image class information. Our minor contributions include the various modules for analyzing the audio, imagery, text, and video from the UI layer, each of which is addressed separately.

II. LITERATURE REVIEW

Content that is unsuitable, and particularly damaging to children takes many online forms. It can mainly be broken down into visual, audio, and textual formats, which can further be broken down into the types of damage that each category can inflict. The most straightforward category to be detected is pornography, which comes in all three of the above mentioned formats, and is a priority of ours here. Another priority is the detection of cyberbullying content, which also comes in all three formats, but is primarily visual and text based. The most insidious and dangerous content that children should be shielded from is sexual grooming and induction to self-harm. We have elaborated on each of these categories of content below, as well as the current methods used to detect, and fight against them. To the best of our knowledge there are no known systems that operate in real time on the UI layer with scarce hardware resources on heterogeneous content types.

A. DETECTION OF EXPLICIT IMAGES

Detection methods for images with pornographic content could be broadly classified as feature-based, region-based and body part-based [19].

Feature-based methods extract the features from the whole image either by encoding local image features (e.g. SIFT) with a visual vocabulary (e.g. [3]) or by learning representations with Convolutional Neural Networks (CNNs). The problem with the first technique is that it depends on hand crafted features whereas the latter could miss some crucial local details (e.g. breast) due to large variations in image backgrounds, scale, scenarios and human poses. Nevertheless, they are widely used and visual vocabulary approach is recognized as low complexity alternative to CNN feature extraction [41].

Region based methods are focused on the detection of regions of interest (ROIs) in which some characteristic of pornographic content is visible. In most cases it is a skin detector that uses a Red, Green, Blue (RGB) model to find the pixels that corresponds to skin regions [18]. Although these methods are not as sensitive to background changes as feature-based ones, the risk of inaccurate skin ROI detection reduce their applicability, as well as their ignorance of any other information presented in the image apart from skin tone.

Finally, body part based detectors first extract previously defined semantic features that describe pornographic content (e.g. breast, belly, bottom) and then use in information as a set of vectors for further classification [24]. The main problem with this approach is its ambiguity and high rate of false positive detections due to small patch support and large appearance variations in the training set. General, fast, CNN based approaches to pornographic image region detection exist, such as [1], but are limited to this category of content.

The especially sensitive category of child pornography can also be addressed using the described pornography detectors without the use of real content from this category in the training set. To do so, the original problem should be split into the problems of detecting general pornographic content and the estimation of the participants' age, respectively [20].

Extracting key frames from video content is outside the scope of our work, but the following aspects should be taken into consideration when evaluating video content in real time. Most existing relevant papers, which train models on the Nucleo de Processamento Digital de Imagens (NPDI) dataset [29], use key frames provided in this data, obtained by using the commercial STOIK video converter [36]. In order to estimate if a video is pornographic or not, the individual estimations from each keyframe are combined by averaging, majority voting or by using a Recurrent neural network (RNN) [41]. Although usage of an RNN increases accuracy by 1 to 2 percent (due to learning long-term dependencies)

and provides streaming video processing capability, it also increases the complexity of the solution. Alternatively, processing each video frame is theoretically possible, but computationally prohibitive.

In order to be able to work with limited computational resources it is important to survey the computational complexity of the state of the art pornography detection methods. The weighted Multiple Instance Learning model [19] can process 5 FPS by using the CPU (Intel Xeon E5-2630) whereas with the addition of a GPU it can provide real time detection of pornography videos at 55 FPS (GeForce GTX Titan X). As the parental control system should protect children from sending as well as from receiving or accessing pornographic content, the preferred solution will be to implement it on users' network equipment (router, access point) or personal devices (desktop, laptop, tablet, mobile phone). The client side implementation of the Not Safe For Work (NSFW) detector is available as JavaScript code which executes in the user's browser [30]. Although it takes a few seconds to initially load a model for a desktop computer, the process of image classification itself usually does not disturb the user browsing experience, but to the best of our knowledge its processing rate for video classification is not reported.

B. DETECTION OF CYBERBULLYING IMAGES

Dadvar *et al.* [8] approached this problem from a non image recognition point of view. Instead of analyzing the images themselves, they studied the content and sentiment of user comments pertaining to the image in question. They further expand on this approach by examining the interaction history of each user that posted comments to the image, and also examined such features as the quantity of capital letters in each comment. This approach does directly detect bullying images, and as such does not eliminate such imagery before harm is caused to the inflicted party.

Bullying imagery frequently depicts vulgar, aggressive, derogatory or otherwise demeaning scenes where a bullying target's face is copied over the victim's face in the original photo. Archives such as [37] attempt to list appropriate methodologies to combat this attack on victims using deep learning, specifically GANs, to detect forged imagery. Several such methods are specifically mentioned that approach the problem of forged imagery, specifically face swapping, and the detection of pornographic content in images and video. Specifically, [19] is the most recent paper in the area of pornographic image recognition, and claims to achieve a remarkable 97.5% accuracy in detecting such cases. It uses a modern YOLO ([2], [33]) type approach to image region classification, which reduces the number of sub-windows to search. Such an approach is also beneficial from the point of view of requiring relatively few training examples in order to be able to robustly produce reliable classifications.

Sabat *et al.* [34] is the first work of its kind that focuses on the image content of memes and attempts to classify them as hateful or not. They manually label a training set of several thousand memes, and restrict themselves to ethnically hateful ones. In other words they limit the hate category to perceived defamation of ethnic groups, as opposed to those targeting individuals in any specific way. The authors extract the text from the meme using Optical Character Recognition (OCR), and derive the semantic meaning of the text using BERT [9]. They extract the visual information of the meme using a VGG-16 CNN, trained using the ImageNet database, and concatenate these features with the semantic textual feature vector from the previous step in order to produce a vector from each meme. They proceed to feed this into a relatively shallow multi-layer perceptron in order to get binary classification results. Please see Fig 2 for details.

This is the basic approach that should be attempted when solving this type of problem, except that it needs to be able to predict hateful memes targeted at individuals, which would require a much more powerful hate predictor, and potentially an image feature extraction module that is capable of detecting deep fakes [37].

C. DETECTION OF CYBERBULLYING IN TEXT

Recent studies report that cyberbullying constitutes a growing problem among youngsters. Studies show that about 18% of the children in Europe have been involved in cyberbullying. In the 2014 EU Kids Online Report [14] it is stated that 20% of children from 11 to 16 years old have been exposed to online bullying. The quantitative research of Tokunaga [38] shows that cyber-victimization rates among teenagers is between 20% and 40%. All of these statistics demonstrate the importance of finding a robust and comprehensive solution to this widespread problem.

A variety of methods have been proposed for cyberbullying detection. Most existing studies have used conventional Machine Learning (ML) models to detect cyberbullying incidents. Recently Deep Neural Network Based (DNN) models have also been applied for detection of cyberbullying. These methods mostly approach the problem by treating it as a classification task, where messages are independently classified as bullying or not. Different DNN models can be used for detection of cyberbullying. The most commonly used are CNNs, Long Short-Term Memory (LSTM), Bidirectional LSTM (BLSTM) and BLSTM with attention. These models respectively vary in complexity in their neural architecture. CNNs are mostly used for image and text classification as well as sentiment classification. LSTM networks are used for learning long-term dependencies. Their internal memory makes these networks useful for text classification. Bidirectional LSTMs, increase the input information to the network by encoding information in both forward and backward direction. BLSTM with attention, gives more direct dependence between the state of the model at different points in time.

D. DETECTION OF INAPPROPRIATE AUDIO CONTENT

Kim and Kim [22] study the problem of detecting objectionable sounds, such as sexual screaming or moaning. This is done in order to classify and block objectionable multimedia

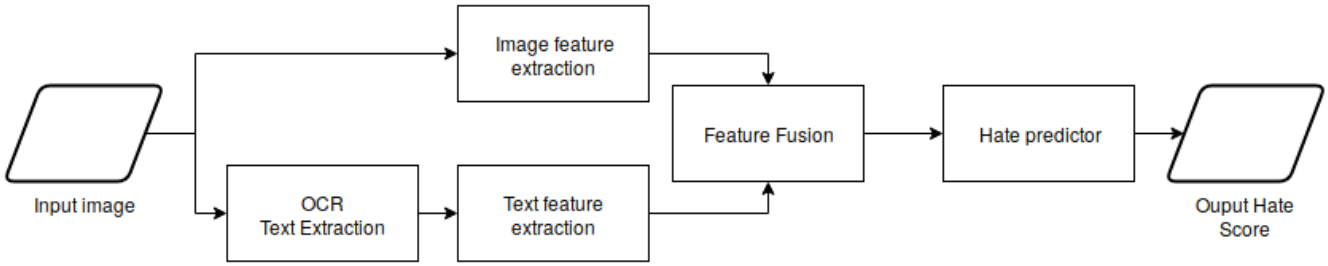


FIGURE 2. The Meme classification model proposed by Sabat [34].

content. They test their method on a database of several hundred objectionable and non-objectionable sound clips, and achieve a precision score of 91.25%. They also found that the audio in pornography is quite distinct and therefore easy to detect. They created spectograms of audio clips using a radon transform for representing the voice, music, and sound effects in pornography.

Some of the most well-known audio processing platforms available on the web were developed as commercial services. Examples include Google Cloud Speech API, which Recognizes 120 languages and variants, Microsoft Bing Speech API, IBM Watson Speech to Text, and Nuance Developers Speech. There are some implementations of speech recognition on web Angular platforms, such as the demo application Web Speech API [16]. There are academic institutions that offer their own solutions, but they often do not offer API access; however, it is possible to develop an environment that would allow calling the libraries they offer through a web API. Examples of these platforms include Kaldi [32], HTK [43], Julius [23], Sphinx-4 [40], and the RWTH ASR Toolkit [42].

E. DETECTION OF ONLINE SEXUAL GROOMING

Online grooming refers to situations when an adult predator wrongfully gains the trust of a child online and then convinces the child to commit sexual acts. Popular social media platforms, gaming sites and child-friendly websites have become breeding grounds for online grooming. These platforms provide access to freely shared personal information on victims that abusers use to find, stalk and bully victims online. As many as 1 in 5 children are solicited sexually while on the internet [27]. Early identification of potential predators who abuse children is critical.

One of the earliest attempts for applying ML to overcome this problem was presented by [31]. He used a weighted k-Nearest Neighbor (k-NN) classifier to recognize predators of underage victims. Later, many other classification algorithms have been used to address the problem, including Entropy-based Classification, traditional Neural Networks and Support Vector Machines. Escalante *et al.* [12] proposed using chained classifiers based on adapting a psychological hypothesis that underscores three stages employed by predators to approach the victim. The work suggests that adopting psycho-linguistic hypotheses can improve classification accuracy. Bogdanova *et al.* [4], enriched the lexical features by adding high-level features such as emotions, neuroticism, and psychological aspects. Also, Cano *et al.* [5] incorporated sentiment features, psycholinguistic features, and discourse patterns as additional features to the original bag-of-words model in order to improve classification accuracy. Ebrahimi *et al.* [10] dealt with the problem from an anomaly detection perspective by using a one-class SVM classification algorithm. Ebrahimi *et al.* [11] learned the word representation internally rather than using general pre-trained word vectors such as word2vec.

III. FRAMEWORK OF THE CASPER SYSTEM

We describe our main contribution here, and give details of its modular parts, based on their roles of analyzing and filtering different types of content. We focus on the audio, image, and text modules, and give details regarding their design and implementation. We also outline the design of the full CASPER system.

A. THE AUDIO MODULE

All audio input and output components, such as the microphone and speakers should be accessible and controllable. The goal is to be able to capture all audio communication, and transcribe it. This transcription is fed into the text processing module, where the analysis of its content is performed. The audio module has no built in analysis, nor classification capabilities, as this is deferred to the text processing module. Kaldi [15] is ideal for this transcription task, as it can be integrated at the operating system level, making these audio signals fully available to our solution. Capturing audio signals is independent from screen capturing, which is why the audio and image analysis tasks are naturally divided. The workflow of the Kaldi software can be seen in Figure 3.

Kaldi is appropriate for the child protection context mainly because it is flexible in controlling all parts of the speech-to-text conversion and could easily adapt to different noisy environments by integrating different acoustic modelling scripts at the operating system level. This approach is also flexible, meaning that it could be used locally, in an edge-computing or cloud-computing architecture.

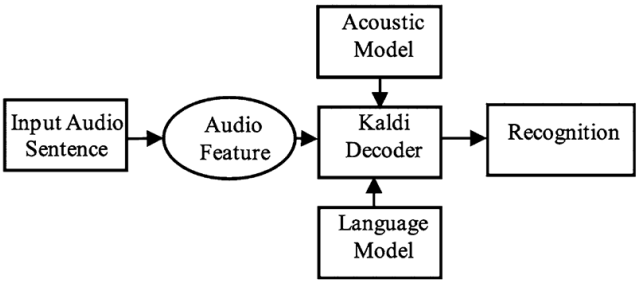


FIGURE 3. The design of the transcription system [15].

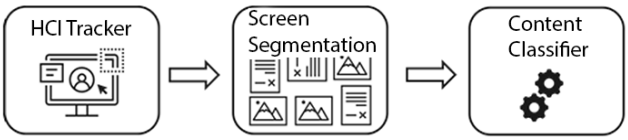


FIGURE 4. The design of the Image Processing Module.

B. THE IMAGE PROCESSING MODULE

The main goals of Image Processing Module are to detect image/video content on the user's screen from the screenshots recorded by the developed HCI Tracker software, and to classify the identified content as appropriate or inappropriate for children. The process is described in Figure 4.

To the best of our knowledge no public dataset exists that is designed to train a model for distinguishing between graphical and textual content, so we assembled a proprietary new Screen segmentation dataset for this purpose. Five students from the Department of Computer Science, University of Beira Interior, Portugal each recorded two 10-minute Internet browsing sessions, that contained mixed (graphical and textual) content. During these sessions, our software saved one screenshot every 3 seconds. Each of the saved screenshots was manually annotated by using the Yolo Mark annotating tool. These efforts resulted in the Screen segmentation dataset which contained 4052 textual and 5967 image areas respectively. The image areas were annotated as containing pornography, nudity, or as being neutral, but this information was ignored for the purposes of training a network that will be able to isolate image areas. Larger image databases exist for objectionable content detection.

The screen segmentation part of the Image Processing Module was based on the Yolo algorithm, and was implemented in OpenCV. This combination of Yolo and OpenCV was chosen as it offers the best inference time for implementing this type of system on CPU [39]. Due to the small size of our labelled dataset, we adopted an approach which is based on fine tuning the publicly available models trained on the Imagenet dataset for object detection.

The content that was recognized as graphical is further processed through this module. The image/video content classifier is designed to use neural network architectures that could be used even on mobile devices in order to reduce computational complexity. The workflow of this module is shown in Figure 5.

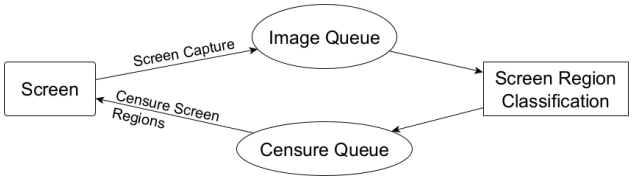


FIGURE 5. The design of the Image Processing Module.

There are three parts of the developed solution. First, there is a module implemented as a worker thread for screenshot capturing. These captures are placed into a queue that is feed into the screen region classification worker. The screen region classification worker consists of the Yolo image region identification step along with the pornography/nudity/neutral classification step described below. Screen regions that are deemed offensive are covered in the censure queue, and ultimately block this content on the screen.

Content recognized as graphical by Yolo is forwarded to the binary classifier that should decide which content is appropriate for children. The dataset for training the classifier was formed using the NSFW Data Scraper [21], which considers only varying degrees of pornograhical content. The acquired dataset was further cleaned in order to remove content that can be considered safe for adults at work, but inappropriate for children. This resulted in 57000 and 46000 images respectively, which can be regarded as pornographic and normal, respectively. MobileNetv2 and MobileNetv3 network architectures were used for this classification step since they can work on devices with limited resources such as mobile phones.

C. THE TEXT PROCESSING MODULE

The main objective of this module is to implement multilingual OCR and cyberbullying detection algorithms. These algorithms should be capable of detecting and recognizing cyberbullying from recorded frames/images/screen shots that contain text written in an arbitrary language. The input to these algorithms are the frames which are basically screen shots from the user's screen. For example, the user is writing messages and comments, reading articles or blogs, is posting/reading on different social media platforms, etc. The text detection and recognition algorithms are capable of detecting the regions with text content that are present on the screen and to recognize the text within text regions. The implemented algorithm for optical character recognition is based on the Tesseract-OCR Engine and has support for over 100 languages. Moreover, it can be further trained and fine-tuned for additional languages.

Cyberbullying detection from text is considered as a multi-class text classification problem. In order to support multiple languages, we decided to use XLM-R which is a transformer based language model, relying on the Masked Language Model objective and is capable of processing text from 100 separate languages, which is very suitable for cross-lingual understanding, a task in which a model is trained in

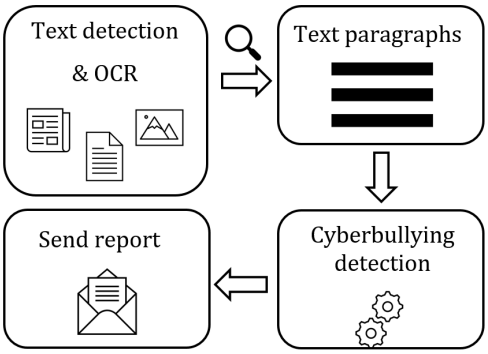


FIGURE 6. The Workflow of the text module.

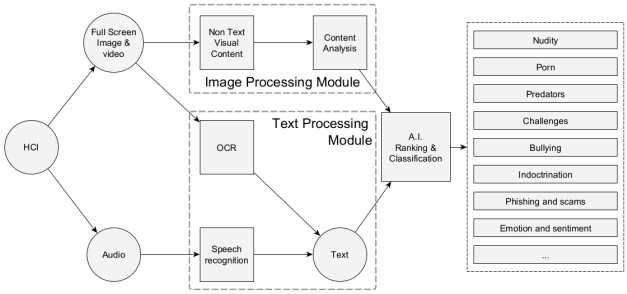


FIGURE 7. The CASPER content analysis and classification workflow.

one language and then used with other languages without additional training data. To further improve the performance of the selected model in the context of cyberbullying detection, we have adopted a scenario to fine-tune the pre-trained xlm-roberta model with specific datasets for cyberbullying detection. We used 10 datasets in total (seven English and 3 Non-English).

The pipeline is depicted in Figure 6. The algorithms within the pipeline are implemented using Python. The recorded frames of the screen, are the input to the multilingual text detection and recognition module. The multilingual text detection and recognition module consists of Tesseract open-source OCR that accepts images as input and detects the regions that contain text as output. Text paragraphs are isolated from these regions and are the input to the module for cyberbullying detection. This module consists of a deep learning predictive model that can be used for inference, which is the process of using a trained DNN model to make predictions against previously unseen data/paragraphs. For each paragraph, a probability of cyberbullying presence is assigned. The system sends an email with a report to the parent depending on the detected presence of cyberbullying. The report contains all of the paragraphs detected as cyberbullying.

For text detection and recognition, we use Tesseract, which is an open-source OCR engine that can recognize more than 100 languages with Unicode support. Also, it can be trained to recognize other languages. Tesseract 4+ includes a new neural network subsystem configured as a text line recognizer. It has its origins in OCRopus' Python-based LSTM implementation, but it is redesigned for Tesseract in C++ called CLSTM. CLSTM is an implementation of the LSTM recurrent neural network model in C++, using the Eigen library for numerical computations. The input image is processed in boxes (rectangles) line by line, and fed into the LSTM model, which produces output. In our implementation we use Pytesseract which is a wrapper for the Tesseract-OCR Engine. The Tesseract-OCR Engine is very fast, as we have obtained an average time of 1.03 sec for recognizing text in an image with 1748 characters in our experiments. We can further speed up the entire process if we replace the tessdata language models with tessdata_fast models which are 8-bit integer versions of the tessdata models.

D. CASPER SYSTEM ARCHITECTURE

The design of our system can be seen in Fig 7. The approach taken by CASPER is to focus all of the processing power on the HCI layer of the user that is to be protected. This means that our system individually analyzes what the user sees and hears (the HCI input) in order to classify content (output). Each of the squares in this figure is a modular task that can be upgraded without affecting the rest of the architecture. Each of the circles represents a modular data collection or amalgamation point. Our work focuses on extracting text via the speech recognition and OCR modules, and visual content by first identifying visually salient non text regions, and analyzing them via the Non text visual content and content analysis modules, respectively. The A.I. module integrates the outputs of the Image and text analysis, as seen in the figure. Alternatively individual modules, such as the content analysis module, can independently reach their own categorizations based on the content they see, and take independent actions, such as blocking certain content, or notifying the children's parents.

IV. EXPERIMENTAL RESULTS

We describe our results individually for the text and image processing modules, as we have customized how each objectionable category of content is handled based on its type. Unless specified otherwise, we used an Intel i7-8565, 8 core machine to conduct our experiments.

A. DETECTION OF INAPPROPRIATE IMAGE & VIDEO CONTENT

Figure 8 shows the CASPER system in action from the point of view of the user. We see a typical search result for 'nude images' on the top of the figure, and what the user would see once this screen was filtered by our system. A description of the deep learning framework that makes this possible is given below.

The dataset for the task of objectionable content detection contained 32190 non objectionable images, and 68313 objectionable images. It was divided into training, validation and test sets, in the ratio of 70%/15%/15% respectively.

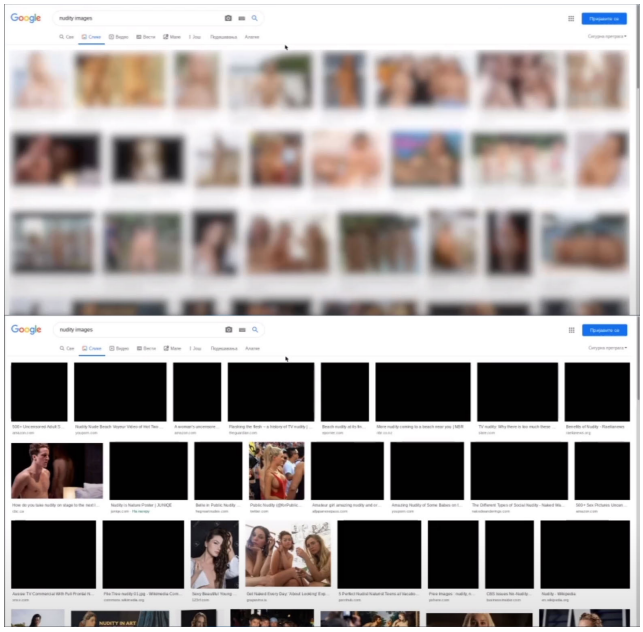


FIGURE 8. A Google search of nude imagery as input into CASPER (Top); The same search, filtered, and given as output by CASPER (Bottom).

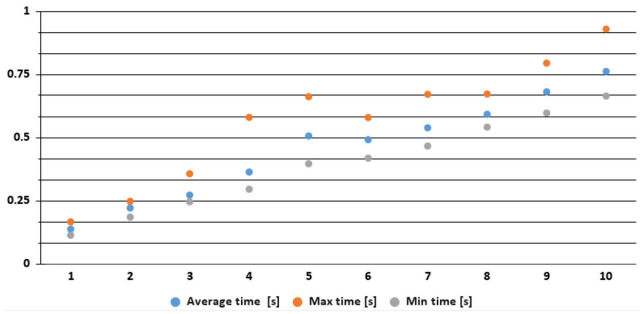


FIGURE 9. Image classification module execution performance.

The Tiny-Yolo3 configuration, specifically designed to detect even small objects, is used since it exhibits relatively modest computational complexity and relatively high accuracy. The screenshots, as inputs to object detection, are scaled to 224 × 224 px for faster inference. Our experiments showed that selected configuration of object detection trained with our Screen Segmentation dataset can achieve 83.7 mean Average Precision when detecting pornographic content. In terms of accuracy both classifiers performed very well (MobileNetv2 – 98%, MobileNetv3 – 97%), however MobileNetv3 significantly improved the speed of image classification.

We analyze each of the three components of the image/video processing module, and achieve 44.9 ms on average for screen capturing. The capture time varied between 27.2 and 76.0 ms. The time performance of the image analysis was measured depending on the number of detected objects in the screenshot while the time performance of censuring was measured related to the number of NSFW objects. The results are presented in Figures 9 and 10.

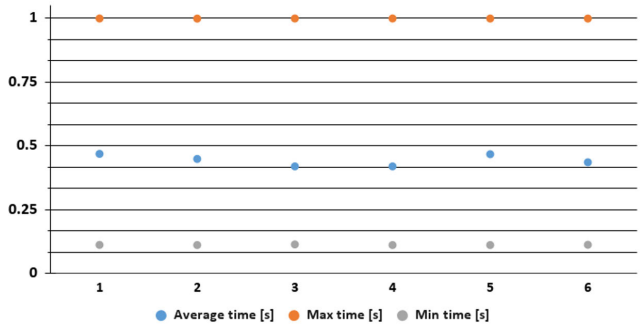


FIGURE 10. Censoring execution performance.

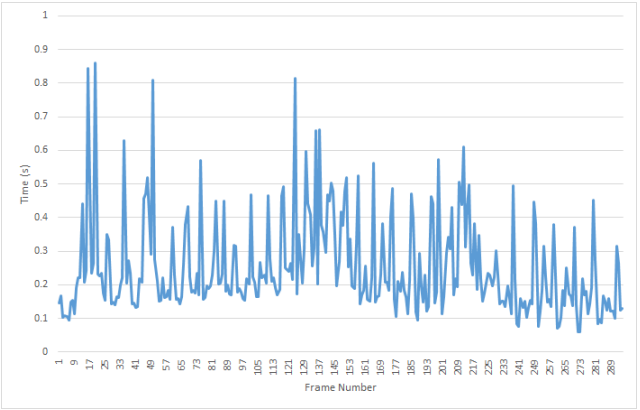


FIGURE 11. Time required to segment the screen into regions of interest.

We can see a linear increase in execution time with respect to the number of detected objects on the screen. However, there is no relation between the number of objects to censure and the censoring execution time, which is expected since this is a simple operation of rendering blank rectangles over certain regions of the screen.

Further analysis of processing times for the image processing module is given in Figures 11 and 12. The system was tested on a single Intel i7 8565U CPU with 20 GB of RAM. The processing time for screen segmentation is on average 0.25 seconds per frame, and the average classification time is 0.15 seconds per region. Based on the current configuration, the system can process 1-2 fps. This performance could be further improved by including the GPU, edge computing, or via algorithm optimization.

B. MULTILINGUAL CYBERBULLYING DETECTION IN TEXT

We consider the detection of cyberbullying in text as a multi-class text classification problem. Multilingual support is especially important in any text classification problem. The ideal case would be to implement a robust algorithm/model that can work with and support different languages, such that adding a new language would be straightforward to implement. The Facebook AI team released XLM-RoBERTa in November 2019 as an update to their original XLM-100 model. XLM-R is transformer based

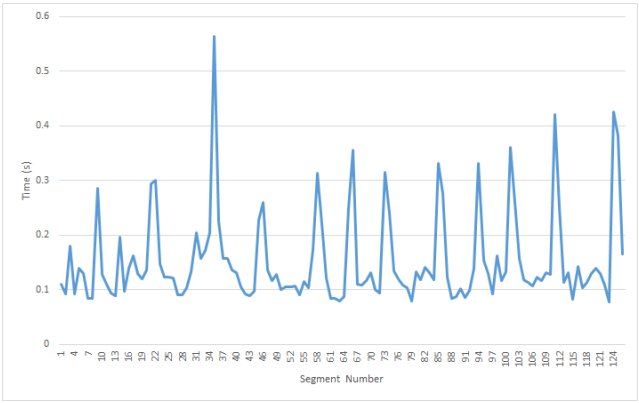


FIGURE 12. Time required to classify the segmented regions within the image processing module.

TABLE 1. Datasets for different cyberbullying categories.

Dataset name	Category	#train samples	#val samples	Language
twitter sexism parsed	Sexism	11,902	2,976	English
youtube parsed	Insult/Hate	2,771	693	English
twitter racism parsed	Racism	10,776	2,695	English
toxicity parsed	Toxicity	127,748	31,938	English
twitter parsed	Racism	13,478	3,370	English
attack parsed	Aggression	92,691	23,173	English
kaggle parsed	Insult	7,039	1,760	English
dkhate	Offense	2,880	720	Danish
offenseval-tr-v1	Offense	25,021	6,256	Turkish
german hatespeech	Hate	375	94	German

language model, relying on the Masked Language Model objective and is capable of processing text from 100 separate languages. XLM-R is trained on 2.5 TB of newly created clean CommonCrawl data in 100 languages. XLM-R uses self-supervised training techniques to achieve state-of-the-art performance in cross-lingual understanding, a task in which a model is trained in one language and then used with other languages without additional training data. This model improves upon previous multilingual approaches by incorporating more training data and languages — including so-called low-resource languages, which lack both extensive labelled as well as unlabeled data sets. To further boost the performance of this model in the context of cyberbullying detection we have adopted a scenario to fine-tune the pre-trained xlm-roberta model with specific datasets for cyberbullying detection. We use Hugging Face's PyTorch implementation. The datasets that were used for fine tuning the pre-trained xlm-roberta model are shown in Table 1.

As we can see from Table 1, different datasets with various sizes, covering different aspects with respect to cyberbullying detection were used. Most of the datasets are in English but we have also selected Danish, Turkish and German. The datasets cover different classes, labels, and categories that can be associated to different aspects of cyberbullying detection such as: sexism, insults, hate speech, racism, harassment, toxicity, personal attacks, aggression and offensive language.

TABLE 2. Predictive performance of the fine-tuned model over the different datasets.

Dataset name	Category	F1 score	Accuracy
twitter sexism parsed	Sexism	0.87	0.91
youtube parsed	Insult/Hate	0.76	0.86
twitter racism parsed	Racism	0.87	0.93
toxicity parsed	Toxicity	0.90	0.97
twitter parsed	Racism	0.84	0.86
attack parsed	Aggression	0.88	0.95
kaggle parsed	Insult	0.84	0.86
dkhate	Offense	0.72	0.89
offenseval-tr-v1	Offense	0.76	0.86
german hatespeech	Hate	0.77	0.89

We have selected BertForSequenceClassification, the pre-trained BERT model with a single linear classification layer on top and fine-tuned it using the datasets from Table 1. The performance of the trained models over the datasets is shown in Table 2. We report the Accuracy and F1 scores, as the accuracy is the most intuitive performance measure, and it is simply a ratio of correctly predicted observations to total observations. F1 Score is the weighted average of Precision and Recall. Therefore, this score takes both false positives and false negatives into account. If we analyze the results, we can conclude that the predictive performance is very good for all of the datasets in general. The average F1 score is also very good for the non-English languages which indicates that the selected model can easily handle other languages as well.

We used a single GeForce GTX 1080 Ti graphics card to train the models. The predictive model accepts arbitrary text as input and assigns classes to it, which include: neutral text or cyberbullying with a corresponding probability. A threshold value can be defined to accept or discard the decision regarding the presence of cyberbullying in the text. The inference process of using a trained DNN model to make predictions against previously unseen data over one paragraph of text on a CPU took roughly 0.04s on average.

V. CONCLUSION

We have shown that our framework is a comprehensive, modular method of identifying objectionable online content. It performs well given limited hardware resources and a limited number of categories of harmful content to detect. Expanding its capability to detect every possible type of objectionable content for children would entail extensive use of the GPU in most circumstances, but is not outside the scope of capabilities of today's state of the art machine learning software.

Currently, our system employs a rudimentary mechanism for reaching decisions regarding content the user sees and hears, where individual modules are able to filter and block screen regions, or notify parents or guardians. Depending on the application, a more sophisticated approach may be required that better integrates the individual modules.

Incorporating online grooming, and self harm detection modules is the main focus of our future work. These new

modules present a challenge in terms of accuracy of detection, especially for online grooming as this is more lengthy attacking process that requires knowledge of previous conversations between parties.

ACKNOWLEDGMENT

The authors would like to thank NGI Trust H2020 EU Project Team for their support, suggestions, and constructive comments. They would also like to thank their colleagues Aleksandar Miljkovic, Petre Lameski, Eftim Zdravevski, and Djordje Hirs for their support during the implementation of CASPER NGI Trust Project.

REFERENCES

[1] N. AlDahoul, H. Karim, M. Abdullah, M. Fauzi, A. Wazir, S. Mansor, and J. See, "Transfer detection of YOLO to focus CNN's attention on nude regions for adult content detection," *Symmetry*, vol. 13, no. 1, p. 26, 2020.

[2] A. Bochkovskiy. (2019). *Darknet*. [Online]. Available: https://github.com/AlexeyAB/darknet

[3] S. Avila, N. Thome, M. Cord, E. Valle, and A. de A. Araújo, "Pooling in image representation: The visual codeword point of view," *Comput. Vis. Image Understand.*, vol. 117, no. 5, pp. 453–465, May 2013.

[4] D. Bogdanova, P. Rosso, and T. Solorio, "Exploring high-level features for detecting cyberpedophilia," *Comput. Speech Lang.*, vol. 28, no. 1, pp. 108–120, Jan. 2014.

[5] A. E. Cano, M. Fernandez, and H. Alani, "Detecting child grooming behaviour patterns on social media," in *Social Informatics* (Lecture Notes in Computer Science). Springer, 2014, pp. 412–427.

[6] A. Chatterjee, K. N. Narahari, M. Joshi, and P. Agrawal, "SemEval-2019 task 3: EmoContext contextual emotion detection in text," in *Proc. 13th Int. Workshop Semantic Eval.*, 2019, pp. 1–10.

[7] M. Dadvar and K. Eckert, "Cyberbullying detection in social networks using deep learning based models," in *Big Data Analytics and Knowledge Discovery* (Lecture Notes in Computer Science). Springer, 2020, pp. 245–255.

[8] M. Dadvar, D. Trieschnigg, R. Ordelman, and F. D. Jong, "Improving cyberbullying detection with user context," in *Advances in Information Retrieval* (Lecture Notes in Computer Science). Springer, 2013, pp. 693–696.

[9] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," 2018, *arXiv:1810.04805*. [Online]. Available: http://arxiv.org/abs/1810.04805

[10] M. Ebrahimi, C. Y. Suen, and O. Ormandjieva, "Detecting predatory conversations in social media by deep convolutional neural networks," *Digit. Invest.*, vol. 18, pp. 33–49, Sep. 2016.

[11] M. Ebrahimi, C. Suen, O. Ormandjieva, and A. Krzyzak, "Recognizing predatory chat documents using semi-supervised anomaly detection," *Electron. Imag.*, vol. 2016, no. 17, pp. 1–9, Feb. 2016.

[12] H. Escalante, E. Villatoro-Tello, A. Juárez-González, M. Montes, and L. Villaseñor-Pineda, "Sexual predator detection in chats with chained classifiers," in *Proc. 4th Workshop Comput. Approaches Subjectivity, Sentiment Social Media Anal.*, 2013, 46-54.

[13] (2010). *EEU COST Action IS0801 on Cyberbullying*. EUCOST. [Online]. Available: https://sites.google.com/site/costis0801

[14] *EU Kids Online: Researching European Children's Online Opportunities, Risks and Safety*, London School of Economics and Political Science, London, U.K., 2014.

[15] J. Guglani and A. N. Mishra, "DNN based continuous speech recognition system of Punjabi language on kaldi toolkit," *Int. J. Speech Technol.*, vol. 24, no. 1, pp. 41–45, Mar. 2021.

[16] M. Hassan. (2016). *Web Speech API Angular*. https://github.com/m-hassan-tariq/WebSpeechAPIAngular2

[17] C. Van Hee, G. Jacobs, C. Emmery, B. Desmet, E. Lefever, B. Verhoeven, G. De Pauw, W. Daelemans, and V. Hoste, "Automatic detection of cyberbullying in social media text," *PLoS ONE*, vol. 13, no. 10, Oct. 2018, Art. no. e0203794, doi: 10.1371/journal.pone.0203794.

[18] F. Jiao, W. Gao, L. Duan, and G. Cui, "Detecting adult image using multiple features," in *Proc. Int. Conf. Info-Tech Info-Net*, Oct./Nov. 2001, pp. 378–383.

[19] J. Xin, W. Yuhui, and T. Xiaoyang, "Pornographic image recognition via weighted multiple instance learning," 2019, *arXiv:1902.03771*. [Online]. Available: http://arxiv.org/abs/1902.03771

[20] J. Jung, R. Makhijani, and A. Morlot, "Combining CNNs for detecting pornography in the absence of labeled training data," Stanford Rep., Stanford, CA, USA, Tech. Rep., 2020. http://cs231n.stanford.edu/reports/2017/pdfs/700.pdf

[21] A. Kim. (2020). *NSFW: Data Scraper*. [Online]. Available: http://github.com/alex000kim/nsfw_data_scraper

[22] M. Jong Kim and H. Kim, "Audio-based objectionable content detection using discriminative transforms of time-frequency dynamics," *IEEE Trans. Multimedia*, vol. 14, no. 5, pp. 1390–1400, Oct. 2012.

[23] A. Lee, T. Kawahara, and K. Shikano, "Julius—An open source real-time large vocabulary recognition engine," in *Proc. EUROSPEECH Scandinavia, 7th Eur. Conf. Speech Commun. Technol., 2nd INTERSPEECH Event*, Aalborg, Denmark, 2001, pp. 1691–1694.

[24] S. Kim, H. Min, J. Jeon, Y. M. Ro, and S. Han, "Malicious content filtering based on semantic features," in *Proc. 2nd Int. Conf. Interact. Sci. Inf. Technol., Culture Hum. (ICIS)*, 2009, pp. 802–806.

[25] Y. Liu, Y. Yang, H. Xie, and S. Tang, "Fusing audio vocabulary with visual features for pornographic video detection," *Future Gener. Comput. Syst.*, vol. 31, pp. 69–76, Feb. 2014.

[26] J. Macdonald and C. Péron, "Artificial intelligence: COMBATING online sexual abuse of children," *Bracket Found.*, to be published.

[27] S. Madigan, V. Villani, C. Azzopardi, D. Laut, T. Smith, J. R. Temple, D. Browne, and G. Dimitropoulos, "The prevalence of unwanted online sexual exposure and solicitation among youth: A meta-analysis," *J. Adolescent Health*, vol. 63, no. 2, pp. 133–141, Aug. 2018.

[28] J. Mallmann, A. O. Santin, E. K. Viegas, R. R. dos Santos, and J. Geremias, "PPCensor: Architecture for real-time pornography detection in video streaming," *Future Gener. Comput. Syst.*, vol. 112, pp. 945–955, Nov. 2020.

[29] (2018). *Nucleo de Processamento Digital de Imagens (NDPI)*. University of Minas Gerais, Pornography Database. [Online]. Available: https://sites.google.com/site/pornographydatabase/

[30] NSFW. (2021). *Client Side Indecent Content Checking*. [Online]. Available: https://nsfwjs.com/

[31] N. Pendar, "Toward spotting the pedophile telling victim from predator in text chats," in *Proc. Int. Conf. Semantic Comput. (ICSC)*, Sep. 2007, pp. 235–241. [Online]. Available: https://ieeexplore.ieee.org/document/4338354

[32] M. Ravanelli, T. Parcollet, and Y. Bengio, "The pytorch-kaldi speech recognition toolkit," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, May 2019, pp. 6465–6469.

[33] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 779–788.

[34] B. Oriol Sabat, C. Canton Ferrer, and X. Giro-i-Nieto, "Hate speech in pixels: Detection of offensive memes towards automatic moderation," 2019, *arXiv:1910.02334*. [Online]. Available: http://arxiv.org/abs/1910.02334

[35] R. Slonje and P. K. Smith, "Cyberbullying: Another main type of bullying?" *Scandin. J. Psychol.*, vol. 49, no. 2, pp. 147–154, Apr. 2008.

[36] (2020). *STOIK Video Converter*. STOIK. [Online]. Available: https://stoik.com/products/video/STOIK-Video-Converter

[37] A. Subin. (2019). *Awesome–Deepfake/Porn Detection Using Deep Learning*. https://github.com/subinium/awesome-deepfake-porn-detection

[38] R. S. Tokunaga, "Following you home from school: A critical review and synthesis of research on cyberbullying victimization," *Comput. Hum. Behav.*, vol. 26, no. 3, pp. 277–287, May 2010.

[39] M. B. Ullah and M. B. Ullah, "CPU based YOLO: A real time object detection algorithm," in *Proc. IEEE Region 10 Symp. (TENSYMP)*, Jun. 2020, pp. 552–555.

[40] W. Walker, P. Lamere, P. Kwok, and B. Raj, "Sphinx-4: A flexible open source framework for speech recognition," *Sun Microsyst.*, to be published.

[41] J. Wehrmann, G. S. Simões, R. C. Barros, and V. F. Cavalcante, "Adult content detection in videos with convolutional and recurrent neural networks," *Neurocomputing*, vol. 272, pp. 432–438, Jan. 2018.

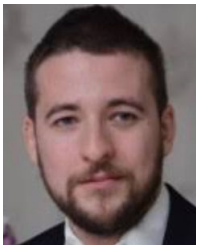
[42] S. Wiesler, A. Richard, P. Golik, R. Schluter, and H. Nez, "RASR–the RWTH Aachen university open source speech recognition toolkit," in *Proc. IEEE Autom. Speech Recognit. Understand. Workshop (ASRU)*, Honolulu, HI, USA, Dec. 2011.

[43] S. Young, *The HTK Book (for HTK Version 3.4)*. Cambridge, U.K.: Cambridge Univ. Press, 2006.

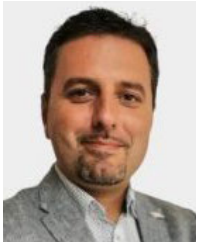
ALEKSANDAR JEVREMOVIC is currently a Full Professor with the Faculty of Informatics and Computing, Singidunum University, Belgrade, Serbia; a Guest Lecturer with Harvard University, Cambridge, MA, USA; and a Visiting Research Fellow with Cyprus Interaction Laboratory, Limassol, Cyprus. He is also recognized as an Expert Level Instructor at Cisco Networking Academy Program. So far, he has authored/coauthored number of research articles and made contributions to three books about computer networks, computer network security, and web development. Since 2018, he has been serving as a Serbian Representative at the Technical Committee on Human–Computer Interaction, UNESCO International Federation for Information Processing (IFIP).

MLADEN VEINOVIC received the B.Sc., M.Sc., and Ph.D. degrees from the Faculty of Electrical Engineering, University of Belgrade, Serbia, in 1986, 1990, and 1996, respectively. From 1987 to 2005, he worked with the Institute of Applied Mathematics and Electronics, Belgrade, where he has been the Head of the Department for Speech Processing, since 2005. He was the Rector of Singidunum University, from 2015 to 2019. He is the author of seven books and a number of journals and conference scientific papers. His current research interests include security, computer networks, databases, and digital signal processing.

MILAN CABARKAPA received the B.Sc. and M.Sc. degrees in electrical engineering and computer science from the School of Electrical Engineering, University of Belgrade, Serbia, in 2009 and 2010, respectively, and the Ph.D. degree in electronics and computer science from the Faculty of Science and Technology, University of Westminster, London, U.K., in 2014. He is currently a Research Assistant Professor and a Teaching Assistant with the School of Electrical Engineering, University of Belgrade. His research interests include software modeling, design and development with the focus on cybersecurity and privacy protection, and design of next-generation communications systems based on 4G, 5G, and the IoT.



MARKO KRSTIC (Member, IEEE) received the Ph.D. degree in electrical and computer engineering from the School of Electrical Engineering (ETF), University of Belgrade. From August 2013 to 2018, he worked with the Regulatory Agency for Electronic Communications and Postal Services (RATEL), as an IT Advisor, and then as a Senior Cyber Security Advisor with Serbian National Computer Emergency Team, from 2018 to 2020. He is currently a Researcher at ETF. He is also part of European Innovation Associate Program, where he is completing his postdoctoral studies. He is a member of COST action CA17124, Digital forensics: evidence analysis via intelligent systems and practices. He has been chosen as a Management Committee Substitute for COST CA18115 action Transnational Collaboration on Bullying, Migration, and Integration at the School Level, as a Serbian Representative.



IVAN CHORBEV is currently the Dean and a Professor with the Faculty of Computer Science and Engineering, Ss. Cyril and Methodius University in Skopje. He has participated in more than 95 scientific papers in journals and conference proceedings, book chapters, and has performed several research stays as a visiting scientist. He is the author of two books. He has been part of or coordinated several national or EU funded research projects . His research interests include combinatorial optimization, heuristic algorithms, constraint programming, web development technologies, application of computer science in medicine and telemedicine, medical expert systems, assistive technologies, knowledge extraction, machine learning, data mining, and learning management systems.

IVICA DIMITROVSKI received the bachelor's degree in computer science, automation and electrical engineering from the Faculty of Electrical Engineering and Information Technologies, in 2005, the M.S. degree, in 2008, and the Ph.D. degree, in 2011. In 2009, he was an Assistant with the Department of Computer Science and Engineering, Faculty of Electrical Engineering and Information Technologies, Skopje. He was a Visiting Researcher with Jožef Stefan Institute, Slovenia, from 2009 to 2010. He is currently an Associate Professor with the Faculty of Computer Science and Engineering, Ss. Cyril and Methodius University in Skopje, Macedonia, and the Head of the Software Engineering Department. His research interests include image processing, medical images, machine learning, content-based retrieval of multimedia data, and web programming technologies.

NUNO GARCIA is currently the Vice-Dean of the Faculty of Engineering and Assistant Professor, University of Beira Interior, Portugal; an Invited Associate Professor with the Universidade Lusófona de Humanidades e Tecnologias, Portugal; and a Researcher with the Instituto de Telecomunicações, Portugal. He is also an Experienced Professor acknowledged in the areas of data acquisition, data fusion, and activity recognition by using mobile devices. His main research interests include ambient assisted living, architectures, algorithms, and platforms for enhanced living environments. He is a member of COST Actions IC1303 and IC1003.

NUNO POMBO (Senior Member, IEEE) is currently an Assistant Professor with the University of Beira Interior (UBI), Covilhã, Portugal, where he is also the Coordinator with the Assisted Living Computing and Telecommunications Laboratory. His current research interests include information systems with a special focus on clinical decision support systems, data fusion, artificial intelligence, software, and software engineering. He is a member with BSAFE Laboratory and the Instituto de Telecomunicações (IT), UBI.

MILOS STOJMENOVIC received the Ph.D. degree in computer science from the School of Information Technology and Engineering, University of Ottawa, Canada, in 2008. He was a Visiting Researcher with Japan's National Institute of Advanced Industrial Science and Technology. He is currently a Full Professor with the Department of Computer Science and Electrical Engineering, Singidunum University, Belgrade, Serbia. He is also a Visiting Fellow with The Hong Kong Polytechnic University, and Riga Technical University. He has published more than 50 articles in the fields of computer vision, image processing, wireless networks, and machine learning, resulting in over 1500 citations, and an H-index of 17. He is also an Editor of three journals, such as *Journal of Multiple-Valued Logic and Soft Computing*, *Ad Hoc & Sensor Wireless Networks*, and IEEE Open Journal of the Computer Society.