A Lightweight Web-Based Filter for Detecting and Moderating Scary Images with CNN

Mohammad Hafiz bin Ismail
College of Computing, Informatics and Mathematics
Universiti Teknologi MARA
Arau, Malaysia
mohammadhafiz@uitm.edu.my

Tajul Rosli Razak
College of Computing, Informatics and Mathematics
Universiti Teknologi MARA
Shah Alam, Malaysia
tajulrosli@uitm.edu.my

Ruzita Ahmad
College of Computing, Informatics and Mathematics
Universiti Teknologi MARA
Arau, Malaysia
ruzitaahmad@uitm.edu.my

Note: This is an early draft. Cite the original article at https://doi.ieeecomputersociety.org/10.1109/ASSIC64892.2025.11158481
.

Abstract*- Online content moderation is important to screen and filter unwanted content on the internet. These may include pornographic, violence, gore, scary and horror imagery. While pornographic image detection and filtering techniques have been studied extensively, very little research have focus on building scary and horror image filter. Thus, this study focuses on developing a web browser-based lightweight scary and horror images filter. Three CNN architectures were evaluated as base classification models candidates. The key metrics for evaluation emphasize model size, inference time, accuracy and sensitivity. The results revealed MobileNetV2 fulfilled all criteria while performing admirably on accuracy and sensitivity in classifying scary images. The model was subsequently quantized and deployed as part of JavaScript implementation on web application. Future works may include expanding the scary images dataset and explainable UI implementation to improve transparency in the model's decision. (Abstract**)*

Keywords—scary images, content moderation, web filter, CNN, deep learning (*key words)***

I. Introduction

The internet facilitates widespread access to diverse multimedia content, including imagery potentially harmful to younger users, such as pornography, violence, gore, and horror. While extensive research has addressed the detection of pornographic and violent content, limited attention has been given to identifying scary and horror imagery, particularly in web-based environments. The advent of Progressive Web Applications (PWAs) has expanded web application accessibility across various devices, including smart televisions, tablets, in-car entertainment systems, and smartphones devices commonly found in households and frequently accessed by children and adolescents. Exposure to horror and scary imagery can negatively impact young viewers by eliciting fear and other adverse emotional responses. Rachman [1] emphasizes that such images can provoke fear and other negative emotions, necessitating tools to mitigate their exposure to vulnerable users.

Livingstone et. al. [2] discovers that children rated scary and gory just below pornographic as content risks. Small children are likely to experience their first fear when viewing scary imagery. The exposure to scary imagery can lead children to develop lifelong phobias and causes sleep disturbance. Children from younger age are also impressionable as they couldn't distinguish fiction from reality and believes the monsters or ghosts from imagery could threaten them in their homes at night [3]. Distorted face as presented on scary monster imagery elicit strong N170 ERP which is visible in brain MRI, compared to normal faces [4]There are cases where young children suffered panic attacks and having nightmares after watching horror theme movie show[5], and this is also true for teenagers respondent when being asked after watching horror movies [6].

Content moderation is a method to ensure unwanted content are filtered or blocked entirely. This method involved content monitoring, analysis and censoring. On web application, image content moderation is typically implemented by blurring the suspected offending images. An example of this implementation is shown in Fig. 1.

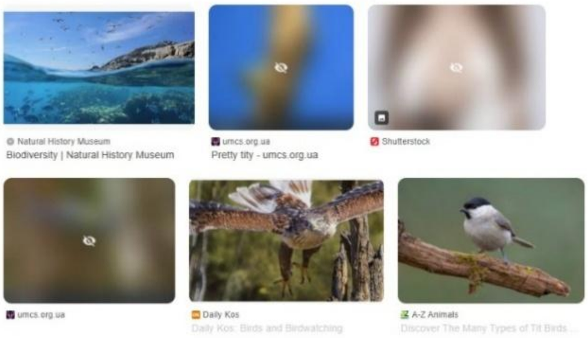

Natural History Museum
Biodiversity | Natural History Museum
umcs.org.ua
Pretty tity - umcs.org.ua
Shutterstock
umcs.org.ua
Daily Kos
Daily Kos: Birds and Birdwatching
A-Z Animals
Discover The Many Types of Tit Birds ...

Fig. 1 Online image content moderation implementation in Google Images.

However, current image filtering techniques which focus on detecting pornography and violence have some limitations when it comes to filtering scary images. Many systems are developed using features like skin-color distribution and explicit object detection that do not capture the subtle and abstract attributes of horror imagery. The models are trained mostly on data that contains explicit content and thus are not well tuned for images that evoke fear through an eerie atmosphere or distorted facial expressions. This results in high false negative rates, where scary content is not detected, and sometimes also false positive results, where non-threatening images are incorrectly filtered. In addition, the design of these filtering methods often ignores the cultural and contextual aspects of what can be considered as scary imagery, which further reduces their effectiveness for protecting younger users.

Therefore in this paper, we proposed a lightweight filter designed to be run on-device which can classify and hide scary and horror themed images on the web. The key contribution of our study are a) Formulation of scary image detection model through transfer-learning, and b) Web-based system for identification and filtering scary and horror themed images. The rest of the paper is organized as follows. In Section II, the related works are discussed. Section III explains the methodology applied in this research, while Section IV details the experimental results. Finally, Section VI concludes the paper and discusses directions for future research.

II. RELATED WORKS

Online pornographic content moderation techniques have been studied throughout the years. Various techniques have been developed to detect online pornographic materials, including text and image content detection[7], skin detection combines with face detection[8], body contour detection [9] body parts detection[8] and local contexts, region-of-interest (ROI) of sensitive body parts [10][11] Pornographic content detection has benefited from well-defined visual characteristics and extensive datasets[12] allowing for the implementation of advanced machine learning techniques and achieving widespread adoption. By contrast, horror content poses unique challenges as there are limited research which focus on online horror image detection.

Li et. al. [13] suggested that the feeling of horror is projected from the relations between regions in the image, thus they proposed a context-aware multi-instance learning model to recognize the relationship between the regions in order to classify horror images. Images can be classified as horror or scary by applying region segmentation and identifying local feature similarities between training and test images. A method for horror image recognition using an emotional attention mechanism has been proposed by [14], which computes emotional saliency maps combined with Bag-of-Words techniques to identify emotional regions based on a user-annotated dataset. A study [15] explored the ability of generative adversarial networks (GANs) to produce fear-inducing images. Participants were asked to classify randomly generated GAN images as "scary" or "not scary. Results indicated that images consistently labeled as scary were generated from specific areas in the latent space, while non-scary images were generated from more dispersed regions. This shows that there are specific attributes which makes a generated image scary. Psychological studies further support this observation, noting humans' irrational fear of darkness and aversion to visual features associated with predatory animals, such as fangs, sharp teeth, and large eyes [16].

On device web image classification can be trained and deployed using pre-built task-specific deep learning architectures, in which CNN model conversion for online deployment can be made possible with TensorFlow.js. Labde and Vanjari [17] demonstrated the successful training and deployment of a deep learning model for skin cancer prediction on the web. Subsequently, a web-based application for plant disease detection and classification was developed using the TensorFlow.js library.

MobileNetV2 is a neural network architecture designed for image feature extraction and image classification tasks for resource constrained environment [18]. This makes the architecture suitable to be used to perform on-device image classification tasks without relying on cloud computing services. MobileNetV2 use cases in edge computing, mobile device and web application. A model trained with MobileNetV2 architecture can be quantized to make it suitable within resource-constrained environment such as web browser. Nguyen et. al. [12] proposed a Google Chrome extension which is able to classify pornographic image and replaced them with warning images. The study compare the performance of Resnet-18, MobileNet, EfficientNet-B0 and GoogleNet in classifying pornographic image. The Google Chrome extension uses RESTful web service concept by sending image URL to the censorship server and the server will use deep learning to classify whether the image is safe. The downside of this method is it still depends on offloading the classification works on remote servers. Additionally, the works only addresses pornographic images filtering problem without addressing horror imagery.

III. METHODOLOGY

The implementation of scary image filter is conducted through three phases: dataset collection, transfer learning, model evaluation and model deployment.

A. Scary Image Dataset Collection

The images for building the scary images filter are collected using image search engine on the internet as there are no standardized dataset for scary or horror imagery. We use Bing Image and Google Image search engine to find scary images. We use search keywords: "scary images", "horror", "ghosts", "zombie", "vampire", "fangs", and "bloody". The sample images are shown in Fig. 2. The images were selected based on the description of fear invoking imagery as described by [1], [4], [14], [19], [20].

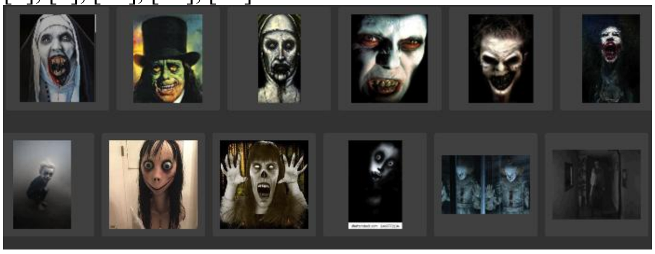


Fig. 2 Sample collected images

It is important to note that we do not claim any copyright to these images, and all were used based on the fair-use policy. These images were used solely for our research purpose to advance understanding and analysis in building the scary images filter. We also use images from the freely available datasets for labeling non-scary images [21]. We've managed to collect 753 scary images and 800 non-scary images. Due to the homogeneous source, the images in the dataset have varied dimensions.

B. Pre-trained CNN Architecture Selection

The pre-trained architectures for this study are chosen based on two criteria:

a) Architectural design : Suitability for deployment in resource-constrained environments, and

b) Compatibility with TensorFlow.js JavaScript tool : Ensuring integration of trained CNN models into the JavaScript library for web-based applications..

Considering these criteria, three pre-trained architectures were selected for evaluating their performance in the proposed scary image web filter implementation: MobileNetV2, EfficientNetV2-B0, and NASNet-Mobile. All three architectures accept input image features of 224x224x3 with pixel values scaled in the range of -1 to 1. Table

It is important to note that MobileNetV3 was excluded as a candidate for transfer learning due to its incompatibility with TensorFlow.js at the time of this study. Specifically, the "Swish" activation function used in MobileNetV3 is not yet supported in TensorFlow.js, making it unsuitable for the intended application.

TABLE I. SUMMARY OF PRE-TRAINED ARCHITECTURES

Architecture	*Parameters*	*Feature Vector*	*Size (MB)*
MobilenetV2	2,260,546	1280	4.8
EfficientNetV2	5,921,874	1280	7.8
NASNet-Mobile	4,271,830	1056	17.2

C. Transfer Learning

Transfer learning is a technique that utilizes well-established deep learning architectures which has been pre-trained with ImageNet weights. Transfer learning process transforms existing deep learning architecture into feature extractors which is then can be trained to adapt to new domain based on new dataset. This reduces the computational requirements and training time compared to training image classification model from scratch.

In the transfer learning process, the convolutional neural network (CNN) layers of the pre-trained models are frozen to preserve the ImageNet weights and prevent them from being updated. This configuration allows the pre-trained model to function as an image feature extractor. To adapt the model to a new domain, the original classification head, which consists of a fully connected layer with 1,000 neurons and a SoftMax activation function, is replaced with a new classification head tailored to classify images as scary or non-scary. By freezing other layers, the training process focuses solely on updating the weights of the newly added classification head.

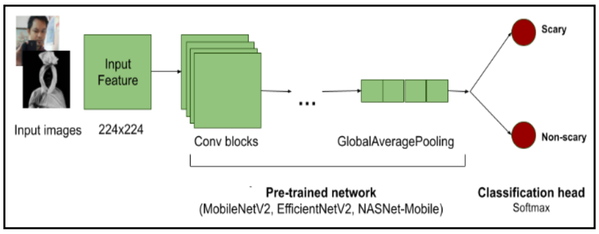


Fig. 3 General Transfer Learning approach

The dataset's training images are partitioned into training and validation sets in an 80:20 ratio. To ensure compatibility with the selected pre-trained CNN architectures, each image is resized to 224×224 pixels.

Data augmentation is applied to the images to improve generalization and reduce overfitting in the resulting CNN models. The augmentation process includes random rotation, zooming, flipping, and image translation. After augmentation, the pixel values are normalized to the range [0, 1], as required by the pre-trained architectures. The images are then fed to each of the pre-trained architectures, for transfer learning process.

Each pre-trained architecture was fine-tuned using the Adam optimizer and categorical cross-entropy as the loss function, with a maximum training duration of 20 epochs. The transfer learning process incorporated early stopping to mitigate overfitting during training. The trained models are then saved using Keras HDF5 format for further processing.

D. Model Evaluation

As the aim of this study is to implement a lightweight Web-based filter, each of the models are evaluated based on classification performance while prioritizing on the inference speed and model size. The performance of each model is assessed using standard evaluation metrics, including accuracy, precision, recall, and F1-score..

Accuracy is the measure of correctly classified images (TP) out of the total tested image. Accuracy provides an overall indication of model performance. However, accuracy usually does not address model's sensitivity which is crucial when building an online filter.

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \qquad (1)$$

Recall (1) is a measure of the model ability to identify all positive instances of scary images. Recall measures the sensitivity of the model. It is crucial to have a web filter model with high sensitivity as a low recall would result in scary images being misclassified as non-scary, potentially exposing users to harmful content.

$$Recall = \frac{TP}{TP+FN} \qquad (2)$$

Precision (3) is a measure of proportion of correctly classified scary images out of all images predicted as scary, which may include false positives (FP).

$$Precision = \frac{TP}{TP+P} \qquad (3)$$

F1-score (4) is the harmonic mean of precision and recall, providing a balanced measure that reflects both the model's accuracy in identifying scary images and its ability to minimize false positives (FP) and false negatives (FN).

$$F1 - Score = 2\ x\ \frac{Presicion\ x\ Recall}{Precision + Recall} \qquad (4)$$

The inference time for each trained models are evaluated using Python environment with Keras machine learning framework. The inference time measurements was conducted on the same hardware to ensure consistency in the evaluation, in which we use a workstation powered by Intel Core i7 14700, with 32GB RAM and accelerated with Nvidia RTX4060ti graphic card.

Each model underwent a warm-up phase to ensure that caching mechanisms and computational graphs were fully optimized prior to measurement. The measurement process involved running five inference iterations with the prepared input before recording any timing measurements. The formula for computing the average inference time is given in (5).

$$T_{avg} = \frac{\sum_{i=1}^{N} T_i}{N} \quad (5)$$

E. Model Deployment

The best performing model is then prepared to be converted to a format compatible with TensorFlow.js for web deployment. The conversion process involved model architecture transformation, weight conversion and post-training integer quantization.

In the conversion process, each Keras neural network layer is mapped to its TensorFlow.js-equivalent layer definition. Furthermore, the model's computational graph is serialized into a format that can be reconstructed within the JavaScript runtime environment.

The model parameters, including weights and biases, are transformed into binary shard files that store their numerical values. During this process, all weights, biases, and activations are quantized into 8-bit integers. The quantization technique involves clustering floating-point values into discrete levels, simplifying the model while enhancing computational efficiency. This optimization improves performance on resource-constrained devices, such as edge computing.

Models	*Accuracy*	*Precision*		*Recall*		*F1-Score*		*Inference time (ms)*
		Scary	Non-Scary	Scary	Non-Scary	Scary	Non-Scary	
MobilenetV2	0.98	0.97	0.99	0.99	0.96	0.98	0.97	41
EfficientNetV2	0.97	0.97	0.98	0.98	0.96	0.97	0.97	48
NASNet-Mobile	0.96	0.97	0.95	0.96	0.96	0.96	0.96	55

TABLE I. PERFORMANCE COMPARISON OF MOBILENETV2, EFFICIENTNETV2, AND NASNET-MOBILE MODELS

IV. RESULTS AND DISCUSSIONS

This section discusses the results from the models' performance evaluation and the prototype of the lightweight scary image web filter on the internet.

Table II presents the performance metrics of three pre-trained neural network architectures MobileNetV2, EfficientNetV2-B0, and NASNet-Mobile evaluated on the classification of scary and non-scary images. The models were assessed based on accuracy, precision, recall, F1-score, and inference time.

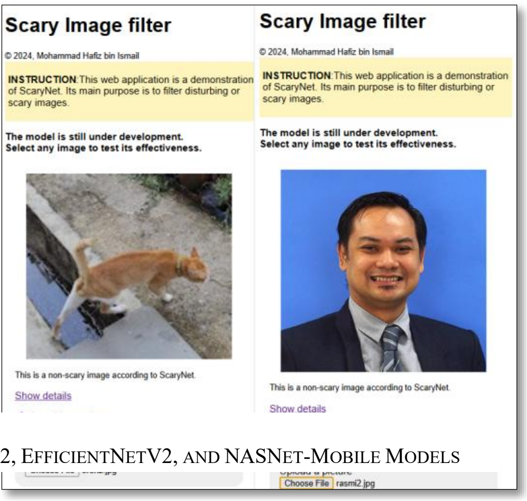
MobileNetV2 achieved the highest accuracy of 98%, followed closely by EfficientNetV2 at 97%, and NASNet-Mobile at 96%. The superior accuracy of MobileNetV2 highlights its effectiveness in capturing the distinguishing features of scary and non-scary images, making it the most reliable model for this task. MobileNetV2 exhibited slightly higher precision for both categories (97% for scary and 99% for non-scary), minimizing false positive rates. This suggests that the architecture effectively identifies scary images without over-predicting.

MobileNetV2 also achieved the highest recall for scary images (99%) and non-scary images (96%), indicating its robustness in correctly identifying all instances of these classes. EfficientNetV2 and NASNet-Mobile showed slightly lower recall values (ranging between 96% and 98%), indicating that MobileNetV2 has a slight advantage in capturing true positive cases. The F1-scores reflect a balance between precision and recall. MobileNetV2 achieved the highest F1-score for both scary and non-scary images (98% and 97%, respectively), indicating a well-rounded performance. EfficientNetV2 and NASNet-Mobile both showed competitive F1-scores (97% and 96%, respectively), though slightly lower than MobileNetV2.

MobileNetV2 exhibited the shortest inference time at 41 ms, followed by EfficientNetV2 at 48 ms, and NASNet-Mobile at 55 ms. The lightweight architecture of MobileNetV2 makes it well-suited for applications requiring fast predictions on resource-constrained devices such as web browsers and mobile platforms.

Based on the evaluation, MobileNetV2 is chosen as the it excels in the performance results, inference time, and the model size. This makes it ideal for deployment in resource-constrained environments like web browsers. Additionally, its compatibility with TensorFlow.js ensures efficient conversion and optimization for Progressive Web Applications (PWA).

Fig. 4 illustrates the scary image web-based filter implemented with TensorFlow.js using converted MobileNetV2 models as its backbone.



Scary Image filter

© 2024, Mohammad Hafiz bin Ismail

INSTRUCTION:This web application is a demonstration of ScaryNet. Its main purpose is to filter disturbing or scary images.

The model is still under development.
Select any image to test its effectiveness.

This is a non-scary image according to ScaryNet.

Show details

Scary Image filter

© 2024, Mohammad Hafiz bin Ismail

INSTRUCTION:This web application is a demonstration of ScaryNet. Its main purpose is to filter disturbing or scary images.

The model is still under development.
Select any image to test its effectiveness.

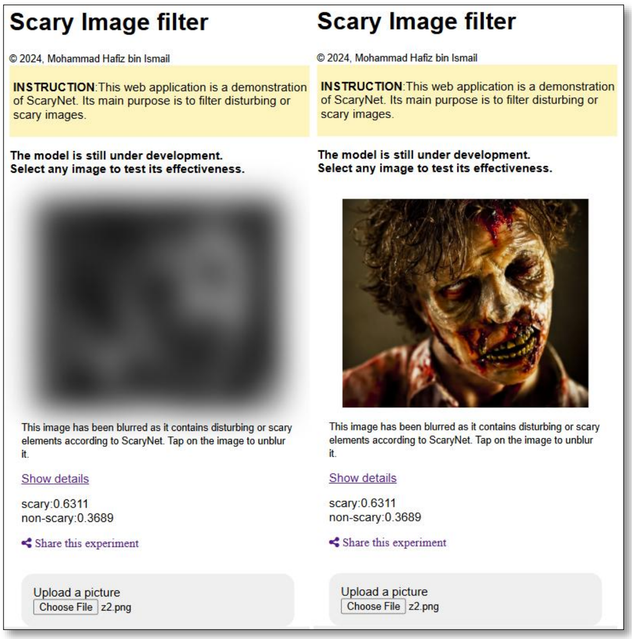
This is a non-scary image according to ScaryNet.

Show details

Fig. 4 Protototype scary image web filter on normal images.

The prototype web filter processes any input image by running inference and returning a list of probabilities indicating whether the image is classified as scary or non-scary. If the image is classified as scary, the filter automatically applies a blurring effect to the image. Users have the option to uncover the image by clicking on it, providing flexibility for manual review. Given that the underlying model demonstrates high sensitivity (i.e., high recall), it occasionally classifies some non-scary images as scary. This behavior, while acceptable for the purpose of prioritizing user safety in an online scary image filter, may result in a few false positives.

To address this limitation, we offer the users to uncover the image and to view the model predicted confidence in image classification.



Scary Image filter

© 2024, Mohammad Hafiz bin Ismail

INSTRUCTION:This web application is a demonstration of ScaryNet. Its main purpose is to filter disturbing or scary images.

The model is still under development.
Select any image to test its effectiveness.

This image has been blurred as it contains disturbing or scary elements according to ScaryNet. Tap on the image to unblur it.

Show details

scary:0.6311
non-scary:0.3689

Share this experiment

Upload a picture
Choose File z2.png

Scary Image filter

© 2024, Mohammad Hafiz bin Ismail

INSTRUCTION:This web application is a demonstration of ScaryNet. Its main purpose is to filter disturbing or scary images.

The model is still under development.
Select any image to test its effectiveness.

This image has been blurred as it contains disturbing or scary elements according to ScaryNet. Tap on the image to unblur it.

Show details

scary:0.6311
non-scary:0.3689

Share this experiment

Upload a picture
Choose File z2.png

Fig. 5. Prototype scary image web filter on scary images.

V. CONCLUSIONS AND FUTURE WORK

This study developed a lightweight online scary image filter, using MobileNetV2 as the core architecture for real-time classification and blurring of scary images. The system achieved high classification accuracy of 98%, with well-balanced precision, recall, and F1-scores. MobileNetV2's fast inference time of 41 ms and compatibility with TensorFlow.js made it suitable for deployment in web-based environments, particularly for devices with limited computational resources such as Progressive Web Applications (PWAs). The high sensitivity of the model ensures effective detection of potentially distressing images, fulfilling the primary objective of enhancing online content moderation. Furthermore, the inclusion of explainability features provides transparency by highlighting the areas of an image that influenced the classification, fostering user trust and understanding of the system's decisions. Overall, the proposed solution offers a practical, efficient, and scalable approach for real-time filtering of harmful visual content.

Future research could aim to expand the dataset to include a broader range of scary and non-scary images, including challenging examples to improve the model's robustness. The web application prototype could also be enhanced with an explainable user interface which can explain the reasoning behind the image classification. This can enhance the users' trust in the system output. The UI can highlight the areas in the image which contributes towards the classification or explain any elements that presents (gore, dark moody background, blood, fangs, etc.) which can contribute towards scary images classification.

ACKNOWLEDGMENT

The authors thank Universiti Teknologi MARA for providing computational resources and colleagues for their support and contributions to this project

REFERENCES

[1] S. Rachman, "The conditioning theory of fearacquisition: A critical examination," Behaviour Research and Therapy, vol. 15, no. 5, pp. 375–387, 1977, doi: https://doi.org/10.1016/0005-7967(77)90041-9.

[2] S. Livingstone, L. Kirwil, C. Ponte, and E. Staksrud, "In their own words: What bothers children online?" 2014, doi: 10.1177/0267323114521045.

[3] G. N. Martin, "(Why) Do You Like Scary Movies? A Review of the Empirical Research on Psychological Responses to Horror Films," Front Psychol, 2019, doi: 10.3389/FPSYG.2019.02298.

[4] C. Gorlini, L. Dixen, and P. Burelli, "Investigating the Uncanny Valley Phenomenon Through the Temporal Dynamics of Neural Responses to Virtual Characters," 2023, doi: 10.1109/COG57401.2023.10333130.

[5] D. Simons and W. R. Silveira, "Post-traumatic stress disorder in children after television programmes," BMJ, 1994, doi: 10.1136/BMJ.308.6925.389.

[6] B. J. Wilson, "Media and Children's Aggression, Fear, and Altruism," Future Child, 2008, doi: 10.1353/FOC.0.0005.

[7] W. Hu, O. Wu, Z. Chen, Z. Fu, and S. Maybank, "Recognition of Pornographic Web Pages by Classifying Texts and Images," IEEE Trans Pattern Anal Mach Intell, 2007, doi: 10.1109/TPAMI.2007.1133.

[8] D. C. Moreira and J. M. Fechine, "A Machine Learning-based Forensic Discriminator of Pornographic and Bikini Images," 2018, doi: 10.1109/IJCNN.2018.8489100.

[9] Y.-J. Park, S.-H. Weon, J. Sung, H.-I. Choi, and G.-Y. Kim, "Identification of adult images through detection of the breast contour and nipple," 2012.

[10] X. Wang, F. Cheng, S. Wang, H. Sun, G. Liu, and C. Zhou, "Adult Image Classification by a Local-Context Aware Network," International Conference on Information Photonics, 2018, doi: 10.1109/ICIP.2018.8451366.

[11] Z. Wu and B. Xie, "Fine-Grained Pornographic Image Recognition with Multi-Instance Learning," Computer systems science and engineering, 2023, doi: 10.32604/CSSE.2023.038586.

[12] D.-D. Phan, T.-T. Nguyen, Q.-H. Nguyen, H.-L. Tran, K.-N.-K. Nguyen, and D.-L. Vu, "LSPD: A Large-Scale Pornographic Dataset for Detection and Classification," International Journal of Intelligent Engineering and Systems, 2022, doi: 10.22266/IJIES2022.0228.19.

[13] B. Li, W. Xiong, and W. Hu, "Web Horror Image Recognition Based on Context-Aware Multi-instance Learning," 2011 IEEE 11th International Conference on Data Mining, 2011, doi: 10.1109/ICDM.2011.155.

[14] B. Li, W. Hu, W. Xiong, O. Wu, and W. Li, "Horror image recognition based on emotional attention," Lecture Notes in Computer Science, 2010, doi: 10.1007/978-3-642-19309-5_46.

[15] P. Yanardag, N. Obradovich, M. Cebrian, and I. Rahwan, "Nightmare Machine: A Large-Scale Study to Induce Fear Using Artificial Intelligence.," in ICCC, 2021, pp. 72–81.

[16] H. C. Barrett, "Adaptations to predators and prey," The handbook of evolutionary psychology, pp. 200–223, 2015.

[17] S. Labde and N. Vanjari, “Prediction of skin cancer using CNN,” in 2022 3rd International Conference for Emerging Technology (INCET), IEEE, 2022, pp. 1–4.

[18] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, “MobileNetV2: Inverted Residuals and Linear Bottlenecks,” 2018.

[19] J. Hamilton, “Monsters and posttraumatic stress: an experiential-processing model of monster imagery in psychological therapy, film and television,” Palgrave Commun, 2020, doi: 10.1057/S41599-020-00628-2.

[20] B. Li, S. Feng, W. Xiong, and W. Hu, “Scaring or pleasing: exploit emotional impact of an image,” in Proceedings of the 20th ACM international conference on Multimedia, 2012, pp. 1365–1366.

[21] G. T. S. P. Limited, “Human Activity Image retrieval.”