Deep Explainable Content-Aware Per-Scene Video Encoding

1st Andy Neparidze
Fraunhofer FOKUS
Berlin, Germany
andy.neparidze@fokus.fraunhofer.de

2nd Ahmed Amine Mchayaa
Fraunhofer FOKUS
Berlin, Germany
ahmed.amine.mchayaa@fokus.fraunhofer.de

3rd Julian David Schäfer
Fraunhofer FOKUS
Berlin, Germany
julian.schaefer@fokus.fraunhofer.de

4th Daniel Silhavy
Fraunhofer FOKUS
Berlin, Germany
daniel.silhavy@fokus.fraunhofer.de

5th Robert Seelinger
Fraunhofer FOKUS
Berlin, Germany
robert.seeliger@fokus.fraunhofer.de

6th Stephan Steglich
Fraunhofer FOKUS
Berlin, Germany
stephan.steglich@fokus.fraunhofer.de

7th Stefan Arbanowski
Fraunhofer FOKUS
Berlin, Germany
stefan.arbanowski@fokus.fraunhofer.de

***Abstract*—In the era of increasing video streaming, optimizing content delivery and finding a good trade-off with minimal effort between quality and size is a critical task. However, varying sensitivities to bandwidth loss across different video content types demand a robust and customized approach to encoding parameters within longer videos containing various scenes. This paper presents a promising novel approach to video encoding, leveraging machine learning to enhance content delivery efficiency, and offers insights into the model's decision rationale. Traditional methods of estimating quality loss through multiple test encodes and interpolation are computationally intensive. To address this challenge, we introduce "Deep Explainable Content-Aware Per-Scene Video Encoding," a machine learning-based approach to video encoding quality prediction given the video scene and encoding parameters. Moreover, we integrate explainable artificial intelligence to enhance our understanding of the model's decision-making process. We have encoded various video scenes using traditional methods and trained our model to learn the relationship between quality loss and specific video contents. We aimed to have our model consider how a specific video scene changes over time; thus, we employed long short-term memory (LSTM) neural networks for predicting quality loss. Our preliminary results demonstrate good accuracy and efficiency, as well as the content awareness of the model.**

***Index Terms*—Video Encoding, Neural Networks, Explainable Artificial Intelligence, Deep Learning, Transfer Learning, Green Streaming**

I. Introduction

The internet is ruled by videos, which make up over 60% of the data flowing through our online world [1]. Moreover, the video quality and careful selection of parameters like the bitrate and resolution for efficient video delivery are very important. Achieving optimal video quality is a complex endeavor that demands meticulous attention and substantial resources, including time and energy, particularly when preparing videos for distribution with varying bitrates and resolutions while meeting specific quality standards that can be measured, for example, using VMAF (Video Multi-Method Assessment Fusion), which will be discussed later. One widely adopted solution in the realm of video encoding is the concept of per-title encoding. This approach involves tailoring the encoding settings for each video individually. When we compress raw video content, we assign a specific bitrate to maintain the desired video quality. The choice of this bitrate is influenced by the video's visual content. In simpler terms, the optimal video encoding is to deliver the highest quality video at the lowest possible bitrate. Factors such as motion, color range, and the distribution of visual elements all play a major role. For instance, a video rich in high-frequency details, with dynamic movements and a constantly shifting camera perspective, is more vulnerable to bitrate reduction than a low frequency, slow-paced, and smoothly transitioning video.

Per-title encoding involves determining the most efficient encoding bitrate for each individual video if possible and encoding the entire video with this bitrate. This chosen bitrate then becomes an important feature to construct an encoding ladder — a set of different bitrate versions of the same video [3]. Encoding ladders are commonly used by video streaming providers to enable adaptive bitrate streaming, ensuring that viewers with varying internet speeds and device capabilities can access content seamlessly. However, a critical challenge emerges when considering this approach: videos are not monolithic entities but rather a collection of distinct scenes, each with its own unique visual characteristics. This inherent diversity makes encoding the entire video with a uniform set of parameters less efficient. This is where the concept of per-scene encoding comes into play. Instead of defining a single

979-8-3503-7099-7/24/$31.00 ©2024 IEEE

encoding profile for an entire video, per-scene encoding takes into account the characteristics of each scene. By doing so, it allows for a more precise evaluation of the visual complexity within each scene and facilitates the application of fitting encoding settings. Yet, this approach comes with its own set of challenges, notably its computational intensity. The prospect of test-encoding every scene in a video to identify the optimal settings is a resource-intensive endeavor. This is precisely where the integration of machine learning into the video encoding process can prove invaluable [4]. Machine learning algorithms can be trained to predict encoding parameters based on the visual characteristics of a scene, offering the potential to replace traditional per-title encoding with a more efficient and adaptive system.

Energy efficiency and sustainability are becoming increasingly important in the production, distribution, and playback of digital media along the entire value chain. More technically efficient video streaming forms the basis for a more energy-efficient and thus more sustainable streaming value chain. Our research focuses on examining major components along the streaming supply chain in terms of their energy efficiency potential, including proof-of-concept implementations of AI-driven sustainable per-scene encoding, green content steering and green streaming approaches for video players utilizing adaptive bitrate formats like HLS and MPEG DASH. Therefore, we see efficient video encoding as one of the key players.

II. Background

Many big companies are working on reducing their internet traffic while still maintaining optimal viewing experience, like Netflix, Facebook, or Google [7]–[9]. The details of their respective approaches differ, but they all take advantage of machine learning to implement their solutions. Hence, this chapter provides a short introduction into the vital components that are often used.

A. VMAF - Video Multi-Method Assessment Fusion

In order to guide the video encoding decisions and assess the quality of these encodings, it is necessary to have a reliable metric of perceived video quality. VMAF is the state-of-the-art video quality metric, developed by Netflix [2], aimed at reflecting the viewer's perception of video quality when streaming content. It is a comparative measure, meaning that it expects two different videos as inputs in order to evaluate their quality against each other. Therefore, VMAF is not an absolute value, rating the quality of videos from poor to excellent, but rather a relative value indicating the quality of one video when directly compared to another video. The goal of VMAF is to bypass the shortcomings of previous video quality measures, like Peak-signal-to-noise-ratio (PSNR) or Structural Similarity Index (SSIM), which are often used during video encoding, even though they fail to consistently reflect actual human perception [2]. To achieve this, VMAF itself utilizes machine learning and has been trained on a large set of videos, including various resolutions, bitrates, and content genres. The quality of these videos was then manually labeled using actual human viewers. Hence, VMAF is able to learn the actual human perception of video quality. On top of that, calculating VMAF is fast and not computationally expensive compared to other traditional video metrics. Because VMAF is open-source, it has been widely adopted since its publication and has been integrated into many third-party video analysis tools, making it very accessible. Therefore, VMAF has proven to be the ideal metric for improving the efficiency of video encoding while still ensuring high perceived video quality.

B. Convolutional and Recurrent Neural Networks

Convolutional and recurrent neural networks (CNNs and RNNs) can be employed for various tasks ranging from object recognition to feature extraction and time series analysis. Since videos are just images (frames) over time, combining CNNs with RNNs is perfectly suited for analyzing video content.

ResNet - ResNet [6] is a powerful and highly influential convolutional neural network architecture for image recognition. It is designed to address the vanishing/exploding gradient problem experienced in deep neural networks. ResNet introduced residual blocks (or skip connections) to allow information to flow more directly through the network. Residual connections allow data to skip multiple layers (see Figure 1), enabling the training of deeper networks without experiencing the degradation problems. As a result, ResNet is very capable at image recognition and has become a fundamental architecture in the field of deep learning. ResNet comes in different flavors, e.g., ResNet-50 and ResNet-101, representing the depth of the respective CNN.

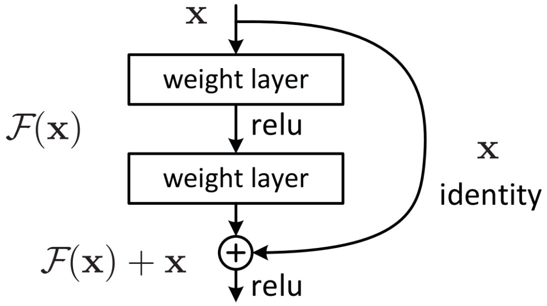
x
weight layer
relu
weight layer
$\mathcal{F}(\mathbf{x})$
x
identity
$\mathcal{F}(\mathbf{x}) + \mathbf{x}$
relu

Fig. 1. A residual building block (taken from [6])

LSTM - Long Short-Term Memory networks are a specialized type of recurrent neural network (RNN) designed to address the challenges of modeling sequential data. Introduced in 1997 by Hochreiter and Schmidhuber [11], LSTMs are equipped with memory cells and gating mechanisms that enable them to capture and retain long-ranging dependencies in sequences. They excel at tasks requiring the understanding of context and temporal relationships, such as natural language processing and time-series analysis. LSTMs are renowned for mitigating the vanishing/exploding gradient problem that is often experienced when training RNNs [18], allowing for effective training on extended sequences.

C. Explainable Artificial Intelligence

Since neural networks can become very deep and therefore act more like black boxes, as in the case of ResNets, it is necessary to be able to explain their decision-making, which is where explainable Artificial Intelligence (XAI) methods come very handy. Many approaches have been proposed [14] and one of the most straightforward is Grad-CAM [13], a technique to generate ”visual explanations” for the decisions of CNN-based models, such as ResNets. Grad-CAM uses the gradients flowing into the final convolutional layer to produce a coarse localization map that highlights the most influential regions in the image for the decision-making of the CNN (see Figure 5).

III. Related Work

A. Green Streaming

Video streaming is a widely utilized online service that consumes a significant amount of energy, contributing to CO2 emissions and environmental concerns. In our prior research [5], we delved into the feasibility and advantages of eco-friendly streaming technologies. These technologies are designed to enhance the energy efficiency and reduce the carbon footprint of streaming content throughout the entire supply chain. Our investigation focused on three pivotal technologies: context-aware encoding, environmentally-friendly media players, and energy-conscious content management. We carried out initial experiments and simulations to assess the effectiveness and influence of these technologies. Additionally, we examined their economic feasibility and potential to generate fresh business prospects within the streaming industry. Our approach was founded on a comprehensive assessment of eco-friendly streaming technologies, offering support to global endeavors in climate action and environmental preservation.

B. Deep Encode

Deep Encode is a versatile machine learning-driven video encoding solution that optimizes bitrate savings while maintaining optimal video quality [23], irrespective of the video codec in use, including H.264, H.265, VP9, and AV1. The previous systems developed by us operated by extracting crucial video features like resolution, frame rate, bitrate, and scene complexity. Subsequently, a machine learning model, trained on a diverse dataset, predicts the best encoding ladder for each video title, determining bitrate and VMAF pairs. Once the model generates predictions, Deep Encode utilizes a standard video encoder to process the video content, producing high-quality videos at significantly reduced bitrates. Notable advantages of Deep Encode encompass consistent bitrate savings of 20-30%, especially beneficial in low-bitrate scenarios, a remarkable reduction in encoding time of up to 50%, and operational efficiency enhancements through automation. Deep Encode includes a feature extractor, a machine learning pipeline to model training and performance improvement, a video encoder, an API for system integration, and a monitoring system. This comprehensive system elevates video encoding efficiency and quality while offering flexibility across codecs.

IV. Proposed Approach

In this chapter, we present a neural network model architecture that is unique in its characteristics because it combines convolutional and recurrent neural networks with explainable AI as well as utilizing transfer learning [10]. Our aim is to demonstrate the feasibility of efficiently encoding a video composed of multiple scenes, taking the video content into account, while closely observing model decision-making for explanations and also considering the time dimension in video scenes. As described in Figure 2 below, our algorithm, in its training phase, takes a video as an input and splits it into scenes; then we generate multiple encodes from the scenes and calculate corresponding VMAF values, then we use the information to train the model, which will be discussed in detail below. To encode a video, we again split this video into scenes, and then we predict the encoding quality for the bitrate value. After encoding every scene, we stitch the scenes together as one video.

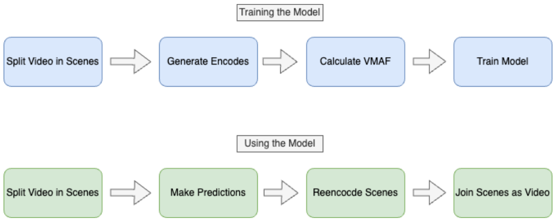

Fig. 2. High level definition of training and the usage of the model

A. Data Preparation

In today’s media world, there are various video codecs, commonly used resolutions and many video types ranging from slowly changing environments to extreme sports and action. In this paper, we reduce the complexity by focusing on videos encoded by one of the most commonly used codecs, H.264 [19]. Also, we narrowed down the video category to only action and nature videos with full HD resolution. This allowed us to demonstrate the model’s capabilities and content awareness with less complexity, after which the model could be extended to various resolutions, codecs, and video types.

We have collected publicly available full HD videos that we pre-processed using FFmpeg [20] and split up into different scenes. Next, the overall 10,000 scenes were split into 80%, 10%, 10% training, test, and validation sets. We generated various encodes for every scene using different percentages of the initial bitrate value (the bitrate of source video without re-encoding with a different bitrate), for example: 100% (the initial value), 95% and so on until 20%; eleven different bitrate encodes per scene in total. Finally, data preparation was concluded by calculating VMAF for every encoded version of the scenes and saving the VMAF values as well as a predefined number of extracted frames from scenes at different timestamps that can be used later for model training. It is noticeable that some scenes are less lossy than the others while re-encoding them with a lower bitrate value. Such videos usually tend to contain low-frequency images or slowly moving camera and objects, as can be seen in nature videos. On the other hand, some types of scenes are very vulnerable to bitrate reduction, as shown in Figure 3 below. Nature scenes tend to retain their quality in most cases when encoded at a lower bitrate, whereas action scenes result in noticeable quality loss.

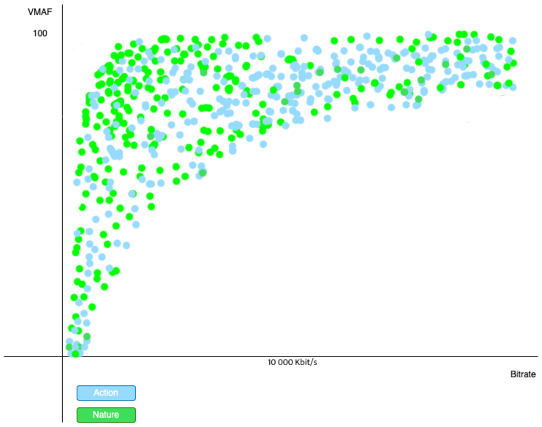

Fig. 3. Bitrate and VMAF value distribution over a couple of hundred differently encoded action and nature scenes

Frames from every scene are used to extract features from them at different timestamps in a scene, and then these features are used to let the neural networks learn the bitrate/VMAF curve for each scene since we have multiple encodes of every scene with corresponding VMAF and bitrate values. Moreover, since we have multiple frames per scene, the model could also learn the temporal relationship in a scene between frames. If we repeat this process for thousands of scenes, the model can generalize well and be able to predict the curve for unseen scenes in a video.

B. Model Architecture and Implementation

The proposed model comprises three major components: feature extractor, where we employ pre-trained ResNet50, on ImageNet Dataset [12], Long short-term memory network that takes ResNet features and the original bitrate of a scene as input and makes a prediction as a form of several VMAF values (predefined number) corresponding to eleven bitrate percentages described above. This gives us the entire bitrate/VMAF curve, so that we can predict the VMAF for any given bitrate value. Moreover, we integrated Gradient-weighted Class Activation Mapping (Grad-CAM) [13] for explanations. As shown in Figure 4 we extract several frames from a scene and feed them into ResNet50 for feature extraction. We obtain a set of 2048 features in a 7x7 format per frame from ResNet. To streamline complexity before feeding these features into the Long Short-Term Memory (LSTM) layers, we employ a Global Average Pooling Layer to obtain 2048 data points and also add the source bitrate of the video to it, resulting in a total input size of 2048 + 1 for LSTM. The initial input layer is directly connected to the first hidden layer, which comprises 2049 LSTM units. Subsequently, the output of this first hidden layer is fed to a second hidden layer, which mirrors the structure with 2049 LSTM units. The second hidden layer, in turn, connects to the output layer, consisting of eleven neurons employing Rectified Linear Unit (ReLU) [15] activation functions. These eleven neurons correspond to the eleven distinct bitrate steps present in our dataset. To reduce the risk of overfitting and enhance model robustness, we introduce dropout [16] layers between the first and second hidden layers, as well as between the second hidden layer and the output layer with a dropout ratio of 0.25 that randomly deactivate some neurons during training.

We decided not to treat the neural networks as a black box but to integrate explainability into the model. Many videos contain logos, watermarks, etc. that can lead to incorrect model training since neural networks sometimes try to find shortcuts by memorizing and associating a logo with a specific decision. But the model should focus on the video content. To address this challenge, we integrated Grad-CAM that leverages the gradients from the model's output back to the final convolutional layer to identify regions within the input that are most influential in determining the model's output. The model is implemented using TensorFlow and Keras [17] in Python.

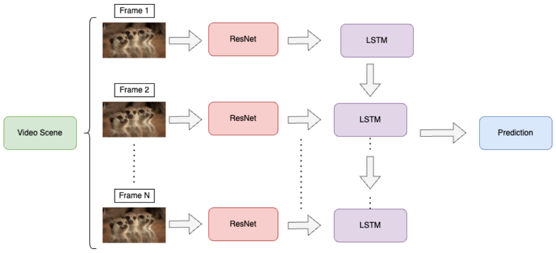

Fig. 4. High level model architecture

V. Model Performance

We encoded over 10,000 scenes, with an equal number of nature and action video scenes, to avoid bias towards a specific category. The model was trained and tested using a Tesla V100-PCIE-32GB GPU. The predictions for scenes, after the model is trained, take several seconds, which is almost instantaneous compared with the standard encoding methods. Then the predictions can be used to encode the video scene with the desired bitrate and quality trade-off. We acquired the model error and standard deviation after testing on videos with higher or equal bitrate reduction than 50% since this is a more realistic scenario in real life, as well as all bitrate reduction variations that can be seen in Table I. During model evaluation, we allow minimal inaccuracies in VMAF prediction since, for example, the difference between encodes with VMAF 80 and 82 is not very noticeable. As a result, more than 70% of all scenes meet the desired VMAF value after encoding using the bitrate from a predicted curve.

TABLE I
MODEL ERROR AND STANDARD DEVIATION

Test Scenes	Std. Deviation	Mean Error	Median Error
Reduction $\geq 50\%$	3.12	2.96	1.77
All Scenes	4.30	3.67	2.32

We have generated explanations for the predictions and observed minimally inconsistent behavior when the model takes unexpected areas of scenes into account, some explanations can be seen in Figure 5, but this topic requires more thorough investigations and will be explored in the future.

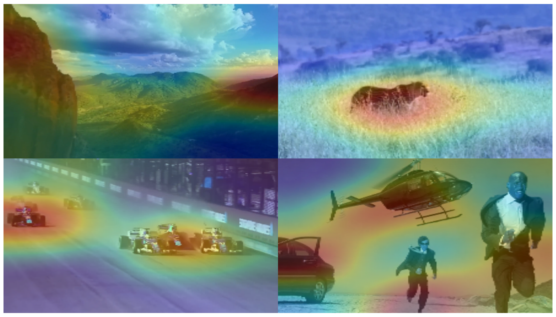

Fig. 5. A heatmap generated using integrated Grad-CAM on nature and action scenes after model training

VI. CONCLUSION

The proposed model demonstrates the feasibility of content-aware video encoding combined with explainable AI. However, it is a proof-of-concept implementation that requires more training and evaluation. This approach could facilitate efficient video encoding. Therefore, we aim to construct a larger dataset for training with multiple video categories and introduce different resolutions to improve the model's capabilities. In addition, the explainability of the model is currently observed based on retrained ResNet outputs. However, it is an essential task to capture explanations through time to answer the question of what temporal changes lead to specific decisions. Also, exploring other explainable AI methods, such as LIME [21], SHAP [22], etc., would give us more insight into the model. In order to obtain a complete encoded video from separately encoded scenes, all scenes need to be stitched back together. It is crucial that the scenes are stitched correctly and that the process is validated at the end so that no frames are missing. Knowing whether the bitrates in the test encodes are optimal is also difficult. Sometimes we can go to a lower bitrate and still get the desired VMAF value, which leads to better model training, but this needs further investigation. Also, reinforcement learning could be an interesting next step, and defining its environment could be challenging.

In summary, we have introduced an interpretable deep learning model incorporating temporal properties. This model exhibits the ability to forecast a VMAF/bitrate curve specific to given video content, showcasing the practicality of content awareness. Our findings underscore the model's advantages in enhancing the efficiency of video coding and its effectiveness in addressing diverse challenges. This study raises several inquiries, and ongoing research is actively exploring potential solutions to these questions.

REFERENCES

[1] NCTA. Report: Where Does the Majority of Internet Traffic Come From?. 10/17/2019. URL: https://www.ncta.com/whats-new/report-where-does-the-majority-of-internet-traffic-come.

[2] Li, Zhi, et al. ”Toward a practical perceptual video quality metric.” The Netflix Tech Blog 6.2 (2016): 2.

[3] J. Ozer. “The Evolving Encoding Ladder: What You Need to Know”. In: Streaming Learning Center (27.5.2019). URL: https://streaminglearningcenter.com/ learning/the-evolving-encoding-ladder-what-you-need- to-know.html.

[4] Seeliger, R., Müller, C. and Arbanowski, S., 2022, November. Green streaming through utilization of AI-based content aware encoding. In 2022 IEEE International Conference on Internet of Things and Intelligence Systems (IoTaIS) (pp. 43-49). IEEE

[5] Seeliger, R., Pham, S. and Arbanowski, S., 2023, June. End-to-end Optimizations for Green Streaming. In Proceedings of the First International Workshop on Green Multimedia Systems (pp. 10-12).

[6] He K, Zhang X, Ren S, Sun J. Deep residual learning for image recognition. InProceedings of the IEEE conference on computer vision and pattern recognition 2016 (pp. 770-778).

[7] Katsavounidis, I., 2018. Dynamic optimizer-a perceptual video encoding optimization framework. The NETFLIX tech blog, 4.

[8] T. Kim, P. Temiyasathit, H. Wang, 2021. How Facebook encodes your videos. Engineering at Meta. 04/05/2021. URL: https://engineering.fb.com/2021/04/05/video-engineering/how-facebook-encodes-your-videos/

[9] F. Mentzer, E. Agustsson, J. Ball'e, D. C. Minnen, N. Johnston, G. Toderici. 2021. Neural Video Compression Using GANs for Detail Synthesis and Propagation. In European Conference on Computer Vision.

[10] Weiss, K., Khoshgoftaar, T.M. and Wang, D., 2016. A survey of transfer learning. Journal of Big data, 3(1), pp.1-40.

[11] S. Hochreiter, J. Schmidhuber. 1997. Long Short-Term Memory. Neural Comput. 9, 8 (November 15, 1997), 1735–1780. https://doi.org/10.1162/neco.1997.9.8.1735

[12] Deng, Jia, et al. ”Imagenet: A large-scale hierarchical image database.” 2009 IEEE conference on computer vision and pattern recognition. Ieee, 2009.

[13] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, D. Batra. 2019. Grad-CAM: Visual Explanations from Deep Networks via Gradient-Based Localization. International Journal of Computer Vision, 128(2), 336–359.

[14] F. Xu, H. Uszkoreit, Y. Du, W. Fan,D. Zhao. 2019. Explainable AI: A Brief Survey on History, Research Areas, Approaches and Challenges. 10.1007/978-3-030-32236-6_51.

[15] Agarap, A.F., 2018. Deep learning using rectified linear units (relu). arXiv preprint arXiv:1803.08375.

[16] Srivastava, Nitish, et al. ”Dropout: a simple way to prevent neural networks from overfitting.” The journal of machine learning research 15.1 (2014): 1929-1958.

[17] Abadi, Martín, et al. ”Tensorflow: Large-scale machine learning on heterogeneous distributed systems.” arXiv preprint arXiv:1603.04467 (2016).

[18] Pascanu, R., Mikolov, T. and Bengio, Y., 2013, May. On the difficulty of training recurrent neural networks. In International conference on machine learning (pp. 1310-1318). Pmlr.

[19] Richardson, I.E., 2011. The H. 264 advanced video compression standard. John Wiley Sons.

[20] FFmpeg. FFmpeg [Computer software]. Available from https://www.ffmpeg.org/

[21] Ribeiro, M.T., Singh, S. and Guestrin, C., 2016, August. ” Why should i trust you?” Explaining the predictions of any classifier. In Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining (pp. 1135-1144).

[22] Lundberg, S.M. and Lee, S.I., 2017. A unified approach to interpreting model predictions. Advances in neural information processing systems, 30.

[23] Silhavy, D., Krauss, C., Chen, A., Nguyen, A. T., Müller, C., Arbanowski, S., ... Bassbouss, L. (2022). Machine learning for per-title encoding. SMPTE Motion Imaging Journal, 131(3), 42-50.