УДК 004.93:004.72
DOI: 10.31673/2412-9070.2025.017844
K. ZDOR, Ph.D. student, assistant;
ORCID: 0009-0008-7640-1499
O. SHALDENKO, канд. техн. наук, доцент;
ORCID: 0000-0001-6730-965X
O. NEDASHKIVSKIY, доктор техн. наук, доцент;
ORCID: 0000-0002-1788-4434
A. MELNYCHENKO, канд. техн. наук, асистент,
ORCID 0009-0000-3588-4772
National Technical University of Ukraine «Igor Sikorsky Kyiv Polytechnic Institute», Ukraine

LEVERAGING VIVIT TRANSFORMERS AND FORESIGHT PRUNING FOR SCALABLE SCENE CHANGE DETECTION ON DISTRIBUTED ARCHITECTURE

Nowadays, the amount of video content is increasing rapidly and requires a more efficient way to analyze it. Scene change detection is a crucial step in video processing because it allows to group of results by context and provides more detailed analyses. This research focuses on the application of Video Vision Transformer (ViViT) architecture to overcome the following challenges - significant computational power requirements and lack of context capturing by convo-lutional and recurrent-based architectures. We also focus on applying foresight pruning for ViViT to reduce resource requirements even more.

By training the ViViT model we achieved a 5.5% improvement in the F1 score for scene detection methods over existing approaches. We also optimized the ViViT model size by 43% and inference time by 10% by applying foresight pruning while maintaining state-of-the-art accuracy.

We also propose a pipeline based on a shot detection algorithm that significantly reduces computational complexity by analyzing only key frames for scene and attribute detection. Applying parallelized processing architecture enables simultaneous scene and attribute detection that leads to a 24.21x speed up against the classic approach of analyzing every frame. This study presents a robust, efficient, and scalable solution for scene and attribute detection that allows a further improvement of methods for scene and attribute detection applications with the potential for real-time analytics.

Keywords: Scene Change Detection, Video Vision Transformer (ViViT), Video Analysis, Parallel Processing, Scalability, Neural Networks, Artificial intelligence, Software Engineering.

Introduction

Nowadays, a massive amount of human-driven content is generated every day, and advanced analyses of this content require a lot of computational power and time. Splitting video into scenes can reduce the amount of heavy calculations and allow detailed statistics to be built. Mathematical methods of scene detection don't have enough accuracy, while neural network approaches have significant computational requirements [7].

The emergence of advanced deep learning architectures like the Video Vision Transformer (ViViT) offers a promising solution by effectively capturing spatiotemporal features in video data while outperforming classical neural network models based on convolutional or recurrent neural networks. While transformer architecture proposes better accuracy and computational costs than other neural network approaches, they still have high computational and resource demands. To optimize the process of detecting scenes and further video attribute extraction, we decided to apply foresight pruning and develop an architecture that allows the parallelized video analysis process and prepares for cloud deployment. As a result, we develop a model that outperforms state-of-the-art results by 5.5% and a pipeline that outperforms a frame-by-frame approach by 24.21x.

Problem Statement

Scene change detection is a complex problem that requires the ability to capture context change. Also, due to the increased amount of generated media data, the need to automate analysis rises drastically. This system must be accurate, scalable, and fast. Traditional methods, like mathematical approaches, show a lack of accuracy. At the same time, neural network solutions based on convolutional neural networks require high computational requirements [11] and have bad accuracy due to a lack of ability to catch temporal dynamics on video. Also, computational resources can grow rapidly with higher-resolution videos, while reducing their size leads to decreased accuracy.

We decided to use Visual transformers for video (ViViT) because this architecture allows us to account for the context of the frames in video content [9]. Transformers can outperform convolutional and recurrent neural networks because of their advanced parallel processing capabilities, but they still have high computational requirements. This can lead to high cost and latency, which can be crucial for analyzing a large amount of content.

To solve these changes, we decided to use an optimized ViViT model to reduce computational requirements and model speed and create an architecture that can also enhance processing speed and reduce resource consumption.

This research addresses the challenge of developing a system that can detect scene change detection efficiently, have high scalability, and be suitable for practical application.

Proposed Approach

First of all, we decided to prepare data before training. Passing full video to the model can lead to several problems. First, due to the complexity $O(n^2)$, long sequences can become a bottleneck and highly increase model complexity and computational time. On the other hand, skipping frames can reduce model complexity, but the model will skip frames and cannot detect the exact frame where the scene was changed. So, we decided to use a shot detection model with 88.9% precision and 88.8% F1 score accuracy [1]. This approach allows us to split the video into shots and compare several frames from each shot. Using this approach, we can reduce the complexity of video analyses by using only essential frames required for scene detection and video attribute analyses. Through the experi-

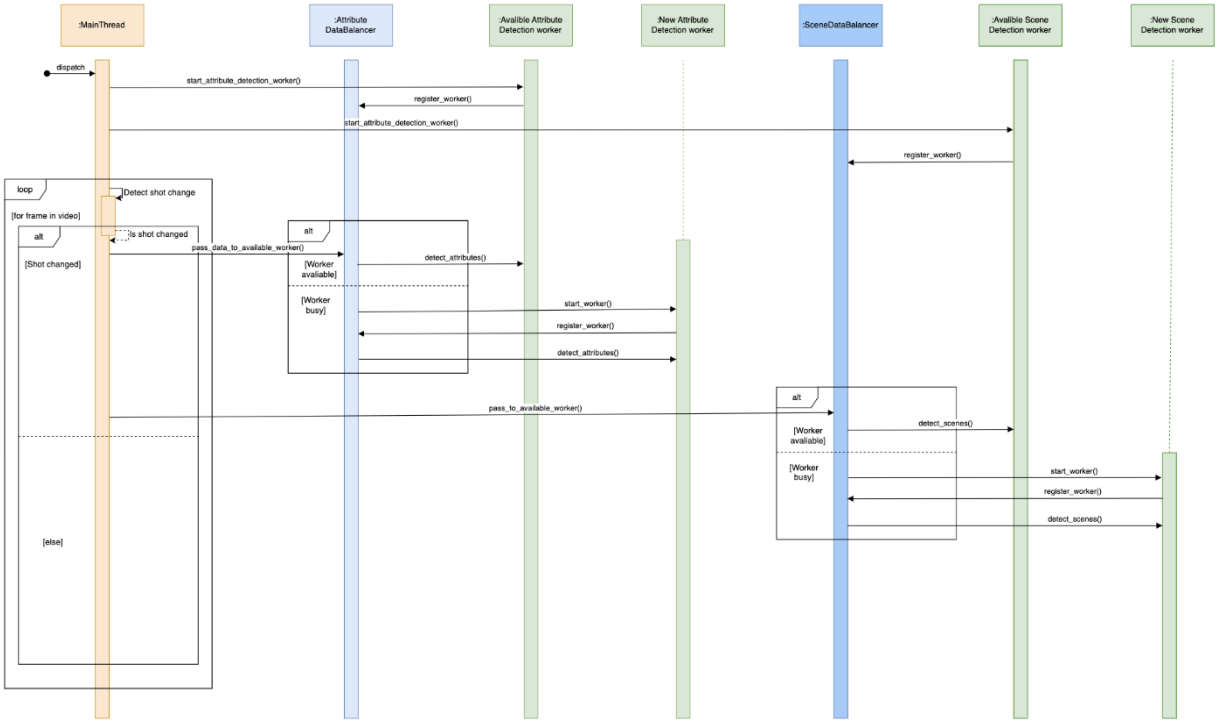

Fig. 1. Sequence diagram of using additional workers for attribute and scene detection

ments, we decided to use four frames to detect scene change. This information collected from these frames contains enough context to detect the scene changes due to ViViT's effective utilization of self-attention mechanisms.

This approach significantly reduces the number of requests for the scene detection model, but the ViViT model requires a lot of computational power. To enhance model efficiency, we decided to use foresight pruning. The foresight pruning algorithm analyzes model architecture and removes redundant and less essential components without substantially affecting model accuracy. After pruning, we also train the model again to fine-tune the pruned model with remained parameters to ensure optimal performance.

At last, we propose an architecture that utilizes parallel processing of video to achieve the best performance. This architecture is designed to work with individual video files as input. The shot detection algorithm requires resizing each frame and then running the algorithm frame by frame. When this algorithm indicates the change of shot, the central frame of the shot is saved to be used in the scene detection part of the pipeline. Additionally, the same central frame is passed to the video attribute inference algorithm. Extracted attributes are written to the database. This inference can also be optimized by automatically storing frames and analyzing them as a batch if algorithms support it. During detecting shots, we start running scene detection, detect the scene change, and record results in the database. As a result, we have an algorithm that splits video into shots and passes required information to other threads that can, in parallel, detect scenes and analyze video attributes, as shown at fig. 1. After the video is analyzed, we can simply request statistics from the database and build a detailed report. The additional benefit of this approach is that it can use both threads and servers (machine hosts) to host the workers. As a result, we can scale parts of the system that require more computational power to avoid bottlenecks.

The benefits of this approach are reducing computational requirements and speeding up the model by applying foresight pruning, achieving state-of-the-art accuracy, and the ability to scale up this algorithm horizontally by deploying it to cloud providers like AWS.

Experiment

We start our experiment by collecting data. We decided to use three different datasets for training because we need to use one of these datasets as a validation set to compare our results with those of our competitors. The Raid, OsVsd dataset was used as a training set, and the BBC Planet Earth dataset was used for validation. As a result, the training set contains 31 videos with 7856 scenes, a total duration of 1047 minutes. The validation set contains 11 videos with 4844 scenes, a total duration of 539 minutes. During the preparation process, all videos were run through a shot detection algorithm, and a mapping that contains shots and scene boundaries was created. It allows us to create a dataset that takes random frames from a sequence of shots and labels if scene changes occur.

After that, we created a ViViT model with four frames as input and a unique data loader that loads positive and negative labels from the created mapping. During training, we apply data augmentation techniques like random resize, crop, color jitter, Gaussian blur, and others. We also randomly swap frames from the same scene and swap scenes. This approach overcomes the problem of the deficient amount of training data and achieves better results and good generalization.

During the experiments, we discovered that the best results can be achieved by passing four frames to the transformer model. Samples are labeled as positive if the first two frames relate to the first scene and the last two frames relate to another scene. This allows the model to focus on getting critical information about shots and comparing them on a contextual level.

The next step is integrating foresight pruning for transformers [2]. By applying foresight pruning, the impact of each weight was evaluated, and based on these results, model inference time was reduced by 10% and model size by 43% while having a low impact on model accuracy. After applying additional training the model achieved an F1 score that is 5.5% better than the state-of-the-art model's (table).

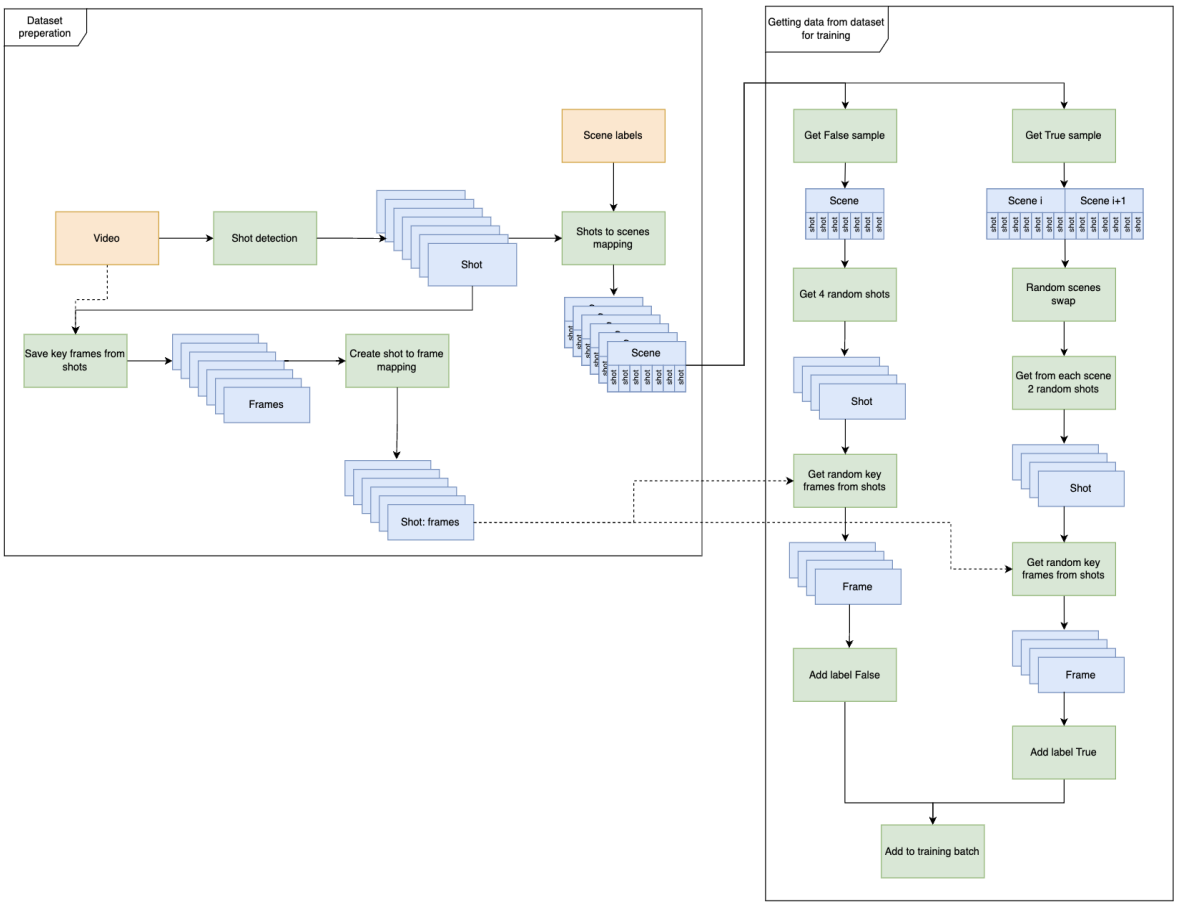

Fig. 2. Example of dataset preparation and selecting data for training

Comparing results of scene detection algorithms

Model	F1
Ours model	0.721
Ours pruned model	0.725
Deep Multimodal Networks [3]	0.67
STG[4]	0.41
NW[5]	0.33
SDN[6]	0.48

As a baseline, we decided to analyze video attributes for each frame. To infer video attributes, we used a face detector[10], face recognition model, and object detection[8]. To test the speed up, several videos of varying lengths were selected, and their frame rate was changed to 24 frames per second. As a result, the baseline speed was 0.95FPS. By using the shot and scene detection model to analyze key frames from shots this approach achieved an 11.14x speed boost(15.58FPS) relative to baseline.

The next step was running scene and attribute detection models as separate processes. To achieve this, we decided to use them as workers that can accept frames and required metadata like timecode, process it, and save it to a shared database. To control threads` workload data balancers were added that create new threads to prevent throttling. This approach was applied to baseline and baseline with shot and scene detection models. The baseline model with this approach shows a 2.17x speed boost against the synchronized baseline. Allying the parallel approach to shot and scene detection models allows us to achieve a 24.21x speed up from the baseline and 12.43x from the parallel version of the baseline (table).

Comparing results of optimization approaches

Approach	FPS	Speed up
Analyze each frame with attribute and scene detection	0.95	1x
Analyze each frame with parallel attribute and scene detection	1.85	2.17x
Shot detection with attribute and scene detection	10.58	11.14x
Shot detection with parallel attribute and scene detection	23	24.21x

Another distinctive feature of this approach is that speedup is expected to be more significant if more resource-consuming models or API calls are applied because long-running calls can be parallelized. Further research can include investigating different feature improvements that can be made for individual models, like applying queuing data and analyzing it as batches, splitting attribute detection models into different processes or duplication these processes, and applying data balancers in case of use models with high execution time. Also, this architecture is exceptionally suitable for transferring it to cloud infrastructures.

Conclusions

This paper explores challenges in creating solutions for analyses of a rapidly growing amount of video content. These solutions should allow for accurate analysis of content, extract meaningful insights, and be suitable for real-world applications. Mathematical approaches for scene detection lack precision and the ability to capture content changes, while neural networks based on convolutional or recurrent architectures are accurate and have good potential for capturing context changes but require significant computational recourses. This research focused on using ViViT transformer architecture with foresight pruning to overcome these challenges. Also, we developed an optimized pipeline that utilizes shot change detection to reduce computational requirements by using only key frames for attribute and scene detection while preserving scene context.

During experiments, we achieved a 72.5% F1 score for scene detection based on ViViT, which is 5.5% better than that state-of-the-art model. We also reduced model size by 43% and inference time by 10%, preserving archived accuracy.

Additionally, the pipeline we created archived a 24.21x speedup boost in processing time compared to the frame-by-frame approach, showing its potential for real-world applications.

Our findings present an accurate, scalable, and available cloud deployment solution for scene and attribute detection that provides an efficient way to analyze video content in different domains.

References

1. Shaldenko, O., & Zdor, K. (2024). Neuro-mathematical fusion for shot change detection in video sequences. Actual Issues of Modern Science. European Scientific e-Journal, 29, 15-24. Ostrava: Tuculart Edition, European Institute for Innovation Development. DOI: 10.47451/inn2024-03-02

2. Melnychenko A., Zdor K. Incorporating attention score to improve foresight pruning on transformer models, Computer Science and Applied Mathematics, 2023, c. 22-27. DOI:10.26661/2786-6254-2023-2-03

3. Baraldi, Lorenzo & Grana, Costantino & Cucchiara, Rita. (2016). Recognizing and Presenting the Storytelling Video Structure With Deep Multimodal Networks. IEEE Transactions on Multimedia. PP. DOI: 10.1109/TMM.2016.2644872.

4. Sidiropoulos, Panagiotis & Mezaris, Vasileios & Kompatsiaris, Ioannis & Meinedo, Hugo & Bugalho, Miguel & Trancoso, Isabel. (2011). Temporal Video Segmentation to Scenes Using High-Level Audiovisual Features. IEEE Transactions on Circuits and Systems for Video Technology. 21. 1163-1177. DOI: 10.1109/TCSVT.2011.2138830.

5. Chasanis, Vasileios & Likas, Aristidis & Galatsanos, Nikolaos. (2009). Scene Detection in Videos Using Shot Clustering and Sequence Alignment. Multimedia, IEEE Transactions on. 11. 89 - 100. DOI: 10.1109/TMM.2008.2008924.

6. Baraldi, Lorenzo & Grana, Costantino & Cucchiara, Rita. (2015). A Deep Siamese Network for Scene Detection in Broadcast Videos. DOI: 10.1145/2733373.2806316.

7. Lin, Weiyao & Sun, Ming-Ting & Li, Hongxiang & Hu, Hai-Miao. (2010). A New Shot Change Detection Method Using Information from Motion Estimation. 264-275. DOI: 10.1007/978-3-642-15696-0_25.

8. Redmon, Joseph & Divvala, Santosh & Girshick, Ross & Farhadi, Ali. (2016). You Only Look Once: Unified, Real-Time Object Detection. 779-788. DOI: 10.1109/CVPR.2016.91.

9. Arnab, Anurag & Dehghani, Mostafa & Heigold, Georg & Sun, Chen & Lucic, Mario & Schmid, Cordelia. (2021). ViViT: A Video Vision Transformer. DOI: 10.48550/arXiv.2103.15691.

10. Mohra, Ashraf & Zakaria, Eman & Mohamed, Wael & Khalil, Abeer. (2019). Face Recognition using Deep Neural Network Technique. ISBN: 9788192958047

11. Shim, Kyuhong & Sung, Wonyong. (2022). A Comparison of Transformer, Convolutional, and Recurrent Neural Networks on Phoneme Recognition. DOI: 10.48550/arXiv.2210.00367.