Automating Film Trailer Generation With Deep Features

Angus Stanton

4th Year Project Report
Computer Science
School of Informatics
University of Edinburgh

2023

Abstract

The trailer is often the first time a viewer is exposed to the plot, cast, mood and style of a film. In a world where short-form video content, on platforms such as TikTok, is being consumed at a meteoric rate, the trailer is more important than ever to capture a viewers attention. As such, it is vital that the elements of a film the viewer is permitted to see within the trailer are those that are best suited for selling the film and making a convincing advertisement. However, selecting these few moments from a film made of hundreds, if not thousands, of individual shots is a challenging task, and is clearly one that would benefit from automation. Many approaches to this have been proposed, with recent methods utilising deep learning for feature extraction and classification showing particular success. In this project we seek to develop and evaluate methods for automating shot selection for trailers. We use the TRIPOD$\bigoplus$ dataset [33], formed of films paired with trailers, and extract a variety of multi-modal deep features. Subsequently, we propose two methods for shot selection: an anomaly detection approach taking inspiration from Movie2Trailer [46], and a supervised approach utilising neural networks to learn to classify film shots as trailer-suitable. Our findings suggest that both methods are capable of producing effective shot selections, however both suffer from limitations that can be addressed with future work.

Research Ethics Approval

This project was planned in accordance with the Informatics Research Ethics policy. It did not involve any aspects that required approval from the Informatics Research Ethics committee.

Declaration

I declare that this thesis was composed by myself, that the work contained herein is my own except where explicitly stated otherwise in the text, and that this work has not been submitted for any other degree or professional qualification except as specified.

(*Angus Stanton*)

Acknowledgements

I would like to thank my supervisor, Mirella Lapata, for her support and advice throughout the year.

Table of Contents

Chapter 1

Introduction

A crucial element of a film's marketing campaign is the trailer, providing audiences with a taste of what to expect from the plot, cast, mood and style present in a film. The rise of internet video has only amplified the importance of the trailer in capturing viewer's attentions. The instant access that online video platforms provide mean that a well developed and distributed trailer can land millions of views in a very short time frame, for example the first official trailer for Marvel's Secret Invasion (2023) obtained 11 million views in just one day on YouTube [9].

Whilst traditional trailers are still a key vector for film marketing, the meteoric rise in short-form video content pioneered by mobile apps such as TikTok has catapulted the importance of quickly digestable, bite-sized content to the forefront of marketing campaigns. With over a billion videos viewed each day on TikTok [8], it is clear that taking advantage of this market is a crucial step to take for succeeding in generating buzz for a film. In addition, streaming services such as Netflix have begun to increasingly rely on short previews of films to entice users into watching recommended films. These are less developed than a trailer, and consist of a small number of shots spliced together to give the viewers an idea about the content of a film.

Whilst trailers and previews present a great opportunity for marketing, their development is not a trivial task. Films contain, on average, around 1000 shots [3] and filtering through these to find effective ones is an unavoidably time consuming job. This problem is further exacerbated by the sheer volume of content available to consumers now, with the streaming service Netflix hosting over 17,000 films as of 2022 [5]. Not only does each film require a preview designed to interest viewers, but the possibility of previews tailored to **user-specific** interests is now also being explored[6]. Generating user tailored previews for 17,000 films is clearly a task that not only would benefit from automation, but requires it.

Automating this task, however, is an inherently challenging problem. Firstly, one must develop a method for **understanding** the contents of film shots such that desirable characteristics can be identified and shots can be selected. This, however, introduces another problem - once shots are processed, the aforementioned 'desirable' characteristics must be identified, either manually or by other means. This is no trivial task, since the key

characteristics of a film shot making it suitable for a trailer or preview are certainly not immediately obvious.

Our goal with this research project is to build upon previous work and explore different methods for selecting film shots for use within a trailer or preview. We split our work into three major sections:

1. First, we explore methods for processing films and extracting salient information from each shot. We use state of the art deep neural networks to extract a comprehensive set of features for each shot with the goal of encapsulating all the necessary information to select shots effectively.
2. Secondly, we build upon previous work to develop an unsupervised system for shot selection utilising anomaly detection on the extracted features. Here, we make the assumption that the 'desirable' characteristic of a shot, making it suitable for a trailer or preview, is that it is in some way different to the other shots within the same film.
3. Finally, we present a supervised method for shot selection, utilising films paired with trailers to train neural networks to predict the 'trailer-suitability' of each film shot. Here, no assumptions are made about the characteristics of a successful trailer or preview shot, and instead we rely on the model to learn these organically.

.

Chapter 2

Previous Work

Automated trailer generation has seen consistent work for many years, with a variety of approaches producing promising results. Whilst several methods have attempted to solve trailer generation with an end-to-end system including the selection and arrangement of shots [24, 51] , many methods reduce the scope of the problem to shot selection or ranking [44, 45]. These methods aim to assist in the trailer generation process by recommending shots to a professional editor, rather than by creating an entire trailer. A system of this sort has already seen commercial use in producing a trailer for the horror film Morgan, utilising shot recommendations produced by an AI system from IBM [45]. Considering the scope of this project, we will mainly focus on the problem of shot selection or ranking.

To find a selection of suitable shots, each previous approach makes its own assumptions regarding the features that make an effective trailer, and indeed what an effective trailer is. Several methods seek to manually identify characteristics that signify a likely trailer shot, then build a model to identify these characteristics and select shots based on this [24, 46]. Other approaches aim to tackle the problem by viewing the trailer generation task as a summarisation problem [20, 33]. These approaches assume that the shots containing the most salient moments within a film are those that are likely to together form an effective trailer. Finally, recent approaches have foregone any assumptions about the characteristics of film shots making them suitable for a trailer, and have instead built supervised models utilising existing trailers to automatically learn the features of an effective trailer shot [49, 33]. All approaches have seen success, with systems from all producing trailers highly rated by viewers.

First, we will consider the view of trailer generation as an effective shot selection problem. Tarbernik et al [46] present movie2trailer (M2T), an unsupervised approach which explores the assumption that the 'non-standard' shots in a film are those that are most suitable for use in a trailer. They investigate the use of anomaly detection techniques for shot selection, using an ensemble of anomaly detection algorithms to identify the 'non-standard' shots. This method proves successful in selecting visually appealing shots and producing trailers which are highly rated by user evaluation. Whilst anomaly detection was shown to be effective in shot selection, vid2trailer (V2T) [24] presents an end-to-end system based on *affective content analysis* (ACA), whereby the

assumption is made that trailer-worthy shots are those that elicit particular emotional reactions in a viewer. Both of these methods produce trailers that are highly rated by human evaluation, however both utilise classical techniques in computer vision and audio processing to extract surface level shot features rather than features that encode a deeper understanding of the content of a shot. Using modern methods to extract more salient information about each shot whilst taking inspiration from these methods is a promising avenue to explore.

Approaching trailer generation from a summarisation viewpoint, PlotsToPreviews (P2P) [20], explores the concept of **preview** generation for films using publicly available plot summaries and further metadata. Scene selection is performed by computing the cosine similarity between embeddings obtained for candidate scenes and plot summary using Sentence-Bert [37]. This method succeeds in retrieving relevant scenes for a preview in a majority of films, however the authors conclude that the method could benefit from using 'audio-visual and high-level semantic information' [20]. Doing just this, Papalampidi et al [33] build upon previous work in Turning Point (TP) prediction [32] (key moments in a film according to screenwriting theory) to aid in shot selection and arrangement for trailers. Incorporating both screenplay and shot level information, the authors extract a variety of multi-model features for each shot using deep neural networks, and combine this information with turning-point based screenplay analysis to inform shot selection and arrangement, producing trailers which score highly in human evaluation.

The previously discussed methods require assumptions to be made regarding the particular properties of a shot that will make it suitable for a trailer. However, recent methods that have shown success focus on learning to rank or classify shots by using existing trailers in a weakly supervised manner, thereby eliminating the need to focus only on certain aspects of each shot and instead **learn** the shot 'properties' that make one suitable for a trailer. CCANet[49] accomplishes this by introducing a paired movie and trailer database, and using a co-attention mechanism to learn soft labels for film shots from trailer shots during the training process. This method obtains state of the art results in the shot selection task without making assumptions about which characteristics signify a shot suitable for a trailer. Whilst CCANet produces promising results, it is limited to only using features extracted on the video modality, and the possibility of expanding upon this method with more informative features is a possible research direction. Furthermore, whilst Papalampidi et al [33] utilise screenplays and turning points for trailer generation, they also show that obtaining impressive results is possible using only shot-level features and a model trained on the binary task of predicting a shots presence in a trailer.

Chapter 3

Dataset and Feature Extraction

3.1 Existing Datasets

Past work in automatic trailer generation can be partitioned, like many machine learning problems, into supervised and unsupervised approaches. Whilst some supervised approaches have been presented [49, 33], the prevalence of unsupervised methods can be attributed to a range of reasons, with a major one being a lack of publicly available datasets to use. Creating datasets for supervised trailer moment detection is an inherently challenging task; Not only is the scale of the data large, with film video files leading to huge datasets, but the labelling process itself is filled with ambiguity - should human annotators manually inspect every shot and rate it's suitability for a film trailer, or is using existing trailer shots for each film enough to encompass the trailer suitable moments from each film? Furthermore, the copyright issues arising from sharing film video files further exacerbates these issues, as datasets that are created are then often impossible to share freely.

CCANet [49] presents one of the first semi-supervised trailer moment detection methods, at the same time authoring the first dataset, the Trailer Moment Detection Dataset (TMDD), for selection of trailer shots. This features 150 films paired with their official trailers, totalling 263,837 film shots and 15,790 trailer shots. These films are split 50/50/50 into Action/Sci-Fi/Drama genres, each split 45/5 for train/test split. Rather than annotating shot labels by hand, they use visual similarity to compute binary labels between film and trailer shots, followed by manual verification of the matches. Using this dataset they obtain state of the art results, showing that a semi-supervised approach pairing films with their trailers is sufficient to accurately perform trailer shot prediction. However, this dataset is not publicly available and as such did not provoke further research.

Papalampidi et al introduce the TRIPOD dataset [32] for movie plot analysis. This is a paired screenplay and synopsis dataset with turning point annotions, and does not feature any video or trailer information. However, in newer work for trailer generation [33], TRIPOD is augmented into TRIPOD$\bigoplus$ with the addition of film videos and corresponding trailers scraped from YouTube.

In our work, we utilise TRIPOD⊕, however we do not utilise screenplay information. This is due to difficulties with aligning screenplay scenes to shots with sufficient accuracy, leading to an inability to utilise screenplay level textual features for usage in shot selection.

3.2 Feature Extraction Background

Once a film is segmented into shots, extracting salient features for use in a selection method is the next challenge. Much of the previous work in the automatic trailer generation literature has relied heavily on traditional audio and image processing techniques to extract shot-level features for use in their methods [24, 46, 51]. Whilst these methods each show success, their feature representations suffer from not encoding an understanding of the content of shots, and instead only encoding the surface level visual or audio features. This has limited the ability of previous methods to choosing shots that are visually or audibly interesting, without considering the content or actions taking place within a shot. CCANet [49] and Papalampidi et al [33] both showcase the impressive results that can be obtained by utilising features extracted with deep neural networks to inform shot selection, emphasizing the benefits of using features that encode a greater understanding of the content of film shots.

Utilising multi-modal (video, audio, text etc) features extracted with deep neural networks is not a new idea in video understanding. Zhu et al, working on using multi-modal deep features for affective content analysis in videos, conclude that ”methods with deep features perform better, and higher accuracy is achieved with the adoption of more advanced DNN feature extractors”. Zhang et al [52] utilise multimodal deep features, including visual and textual modalities, for relationship and interaction analysis of movies, further cementing the use case for multi-modal deep features in video understanding. Whilst neither of these works are in the trailer generation domain, Palampidi et al back up our hypothesis that multi-modal deep features will help with shot selection, showing that using multi-modal features is an easy way to outperform models such as CCANet which only uses image features.

As for the methods used for feature extraction itself, different modalities have had different models present themselves as the de facto standard in feature extraction for each domain.

In the field of image processing, the introduction of deep convolutional neural networks (CNNs) has led to a huge improvement in the performance of neural networks on image based tasks, including image classification [27] and object detection [13]. A key part of this success can be attributed to deep neural networks ability to automatically learn rich feature representations of their inputs to be used downstream. Networks trained on datasets such as ImageNet [16] have been shown to dramatically boost performance when their representations are used as inputs to models for new tasks [23], improving performance and lowering the amount of required training data. Previous approaches in trailer shot selection, such as CCANet, rely on features outputted by ResNet [22] based architectures trained on ImageNet. Whilst these have shown impressive performance on a variety of video understanding tasks, newer models such as CLIP [36] from OpenAI

promise even more informative image encodings. CLIP is trained on an image-caption pairing task and produces image embeddings that have been shown to outperform those from ImageNet trained ResNet models on many vision-and-language tasks [43], hopefully also translating to more informative features for trailer shot selection.

Whilst deep CNNs have been revolutionary for image understanding, we are dealing with the video domain and as such utilising methods for understanding actions throughout time is crucial for our task. A large variety of methods have been proposed for this, and we will focus on the usage and development of 3D CNNs. 3D CNNs [47] augment the traditional CNN architecture with one crucial detail – filters also process a temporal window across adjacent frames, thereby learning spatiotemporal relationships. Building on this work, the SlowFast [19] network dramatically improves results by processing two distinct aspects of video. It utilizes a 'slow' pathway, dedicated to extracting semantic information from, and across, each frame at a relatively low framerate, and a 'fast' pathway dedicated to processing motion with a much higher temporal resolution.

Subtitles, when available, can be crucially informative in video understanding. With the advent of the transformer model [48], deep learning for natural language understanding has seen a surge in effectiveness. Large pretrained language models, such as BERT [18] have advanced the state of the art natural language understanding, inference and generation. Using these large pretrained models to generate sentence level embeddings for use in downstream tasks has been shown to be effective, with embeddings from Sentence-Bert [37] obtaining state of the art performance whilst providing fast inference times.

Pretraining large neural networks on large amounts of generic labelled or unlabelled data followed by finetuning on a smaller amount of task-specific labelled data is a paradigm that has proved very successful for applying deep neural networks to problems with a relatively small amount of labelled training data. This has been used in image processing, where networks pretrained on ImageNet are finetuned for different downstream tasks, and in natural language processing, where large language models pretrained on unlabelled data are then finetuned on task specific labelled data. However, whilst finetuning is an attractive option, it has been shown that for many tasks only training the classification head, known as linear probing, is sufficient to deliver performance on par with, or only slightly worse than, finetuning for both text [34] and image [35] modalities. Therefore, in this project we only consider the usage of pretrained models for feature extraction, rather than for finetuning as it is much more resource efficient with comparatively small performance impacts.

3.3 Data Processing

3.3.1 Feature Extraction

We utilise a number of features for each shot, across a variety of modalities and timescales, including single frame features, video motion features, audio features, subtitle based textual features and facial emotion features. Before feature extraction, films are first segmented into shots using PySceneDetect [7], and subtitle timestamps

are used to align them with the segmented shots. Shots of less than 100 frames are discarded, as they are too short for processing and displaying in a trailer. An overview of the data processing pipeline is shown in Figure 3.1.

A film shot is a collection of still images played in sequence, and as such the features extracted need to represent both the content of each frame, and what is happening **across** frames. To extract frame level features we sample one in every ten frames and feed each through two different networks; The first is a ResNet network pretrained on ImageNet for image classification, the second is CLIP. The penultimate layer embeddings from both of these networks are saved and mean-pooled across all sampled frames. For temporal video features, we utilise SlowFast [19] pretrained on the Kinetics dataset for human action recognition, again utilising the penultimate layer output as an embedding vector.

Whilst trailers often do not directly feature the original audio of a shot, instead opting to overlay music or narration, subtitles can provide information about a shot that is hard to obtain using only video. To obtain subtitle textual features, we utilise Sentence-Bert [37] to extract sentence level embeddings from the subtitles and mean pool these across all the utterances present in a shot.

Furthermore, alongside the subtitles, a shot's original audio is also assumed to be informative in classification. We extract shot-level audio features using YAMNet [2] trained on Audio Set [21] to classify audio clips into 521 classes. We input the entire audio segment from each shot to the network, and save the final activations as audio features.

Taking inspiration from the work of vid2trailer [24], which utilises affective content analysis for shot selection, we make the assumption that utilising the emotions present on faces within a shot will be an effective input feature to our model. To obtain emotion features, we first sample one frame per second for each shot to use for facial recognition. For each frame, we utilise RetinaFace [17] to extract face bounding boxes and facial landmarks such as eye positions. This system is capable of recognising faces at extremely low resolution, and very high rotation angles relative to the camera. As such, it tags many faces that are a) too small or b) too turned away from the camera to effectively determine an emotion. We therefore implement two filtering stages; Firstly, bounding boxes with any side smaller than 80 pixels are filtered out. After this, any detected faces where the eyes are too close together relative to the height of the bounding box are filtered, as this is an effective metric to determine if a face is rotated too much away from the camera. Finally, for feature extraction, we utilise HSEmotion [40, 41, 42], a model used for emotion recognition detection and feature extraction. For each face that passes the filtering stage, we extract features using the HSEmotion model on the cropped frame image. For the final shot facial emotion features, we mean-pool all the emotion feature vectors across all extracted faces.

For extracting the visual features, including CLIP, ResNet and SlowFast, we utilise the HERO Feature Extractor [30] repository. This was previously used in work for video and language understanding with multi-modal features.

The dimensionality of each of the extracted features is shown in Table 3.1.

Group	Feature	Dimensionality
Visual	CLIP	512
	ResNet	2048
	SlowFast	2304
Linguistic	Sentence-Bert	384
Audio	YAMNet	521
Emotion	HSEmotion	1280

Table 3.1: Feature Statistics.

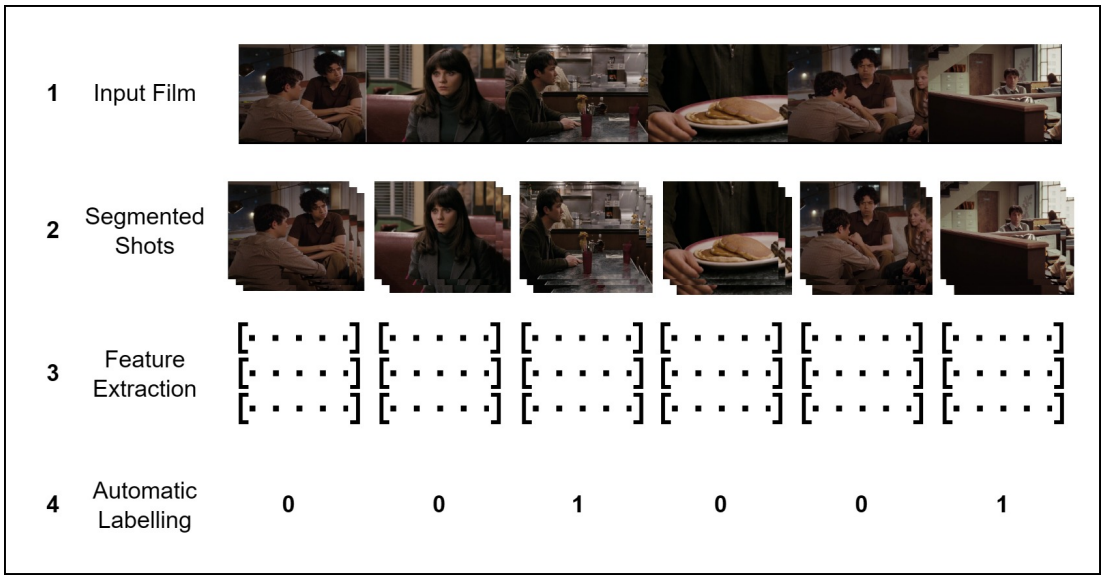

Figure 3.1: Data processing pipeline.

3.3.2 Automatic Labelling

To both evaluate and train models for trailer shot selection, shot level annotations indicating the presence of a shot in a trailer are needed. These can be obtained by hand, with human annotators manually comparing film and trailer shots, however this is an incredibly labour intensive task, requiring resources beyond the scope of this project. Instead, silver standard labels are utilised. These are obtained by computing the similarity between film shot features and trailer shot features. In particular, the cosine similarity between the visual shot and trailer features are used, and the highest similarity film shot is given a positive label. A similarity threshold of 0.85 is set, as a trailer often features many shots or visuals that do not feature in the original film.

3.4 TRIPOD⊕ Summary

The statistics for the TRIPOD⊕ are shown in Table 3.2. In the train and development sets, films are paired with a number of trailers found online, leading to an average of around 30 positive trailer labels per film. In the test set, however, each film is paired with only its official trailer, leading to an average of only 18 positive labels per film. Subsequently, in the train and dev set the average percentage of positive labels per film

TRIPOD⊕	**Train**	**Dev**	**Test**
No. Films	84	38	41
No. Shots	81,400	34,100	48,600
No. Trailers	277	155	41
Avg. Trailers per film	3.3	4.1	1.0

Table 3.2: Dataset Statistics.

is 8.3%, and in the test set the average percentage is only 4.2%.

Chapter 4

Anomaly Detection

4.1 Background

Anomaly detection, often referred to as Outlier Detection, refers to the task of 'finding patterns in data that do not conform to expected behaviour' [14]. It is an area that sees applications across a variety of domains and modalities, from classical problems on tabular data such as intrusion detection within computer networks [25] and fraud detection [10] to newer applications on a variety of modalities, such as surface defect detection in images [29].

A large variety of methods for anomaly detection have been proposed, with the effectiveness of each approach varying wildly depending on the type of data and its distribution. Typically, methods involve building a model that expresses the properties of regular data, and then using this to identify unusual, or 'anomalous', data. . Whilst methods for anomaly detection are often supervised or semi-supervised (models using only data labelled as 'normal'), many of these can be adapted to the unsupervised domain. Unsupervised methods make the assumption that 'normal' instances are far more prevalent than the anomalous ones, meaning that an accurate model of the regular data can be made from samples which contain anomalous instances. We will explore several options for unsupervised anomaly detection when developing our system.

Diving deeper into previous work in anomaly detection on features extracted with deep neural networks, we see many publications dealing with the image modality [12, 38, 31, 50], but relatively few from other domains. We will look to apply methods used in the works focused on the image modality to all the multi-modal features in our dataset discussed previously. Exploring the methods the previous work employs, we see a variety of existing anomaly detection methods proving effective for use on deep features. K-Nearest Neighbours (kNN), an algorithm for both supervised and unsupervised anomaly detection which uses the largest distance to each data points K neighbours as an anomaly score, is shown to work effectively on features extracted by networks pretrained on ImageNet in both the static image defect detection domain [12] and the continuous video surveillance domain [31]. In particular, the success of kNN on the video surveillance problem is a result that will hopefully effectively translate to the anomalous shot detection task, both of which deal with a continuous video domain.

Both One-Class Support Vector Machine (OC-SVM) [50], an unsupervised method that fits an SVM to the data and uses the hyperplane distance as a anomaly score, and Multi-Variate Gaussian [38] are also shown to be effective in the static image defect domain. The variety of effective methods is a promising sign that employing the same ensemble technique used in M2T, which combines 8 different anomaly detectors, will be an effective route to take.

As explored in our Previous Work chapter, utilising anomaly detection as a solution to the effective shot selection problem is not a new idea. Movie2Trailer (M2T) [46] makes the assumption that the shots within a film that are suitable for a trailer are characterised by being in some way 'different' from the other shots within the same film. To exploit this assumption, anomaly detection is performed on surface level visual and audio features in the hope of finding visually and audibly interesting shots. This approach proves effective, producing shot selections and trailers that are highly rated in viewer evaluations. However, whilst effective, this approach does not make use of an understanding of the content of shots, instead focusing on surface level shot features. The success, and limitations, of M2T, when combined with the success of anomaly detection on deep features gives a good indication that transferring the techniques utilised in M2T to the deep features we have extracted is a promising direction to take. For the rest of this chapter we will develop and evaluate a method for shot selection utilising anomaly detection on deep features.

4.2 Method

4.2.1 Overview

The overall architecture of our anomaly detection system is presented in Figure 4.1. It is similar to that used in the M2T, with a few differences. A key aspect of the system is to process each film in isolation, finding the most anomalous shots with respect to only the shots from the same film, rather than across all films.

First, shot level features are extracted with the methods described previously. These include frame, motion, audio, subtitle and facial emotion features. Each feature, across all shots, is used in isolation of the other features until the final combination stage. Features are pre-processed before anomaly detection, involving normalization and dimensionality reduction. Each feature, across all the shots, is then run through a selection of anomaly detection methods, each outputting an anomaly score for each feature and shot pair. These scores are then combined across all features and all detectors with an ensemble method to produce a final score. Finally, the shots with the top K scores are selected for use in the trailer.

4.2.2 Pre Processing

The high dimensional deep features extracted for each shot present a challenge for anomaly detection [56]. Nazare et al perform a comprehensive analysis of the effect of normalization and dimensionality reduction using principle component analysis (PCA) on the performance of anomaly detection on high dimensional features from

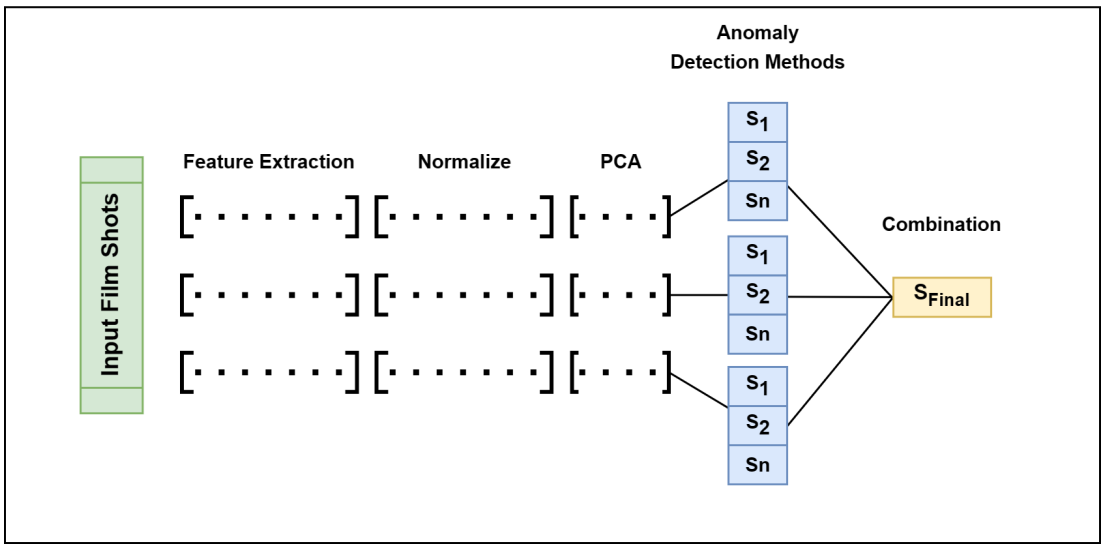

Figure 4.1: Anomaly Detection Model Overview

a deep neural network, finding both to have significant effects on results. As such, we implement the same [0-1] normalization scheme they show to be effective, and experiment with a variety of PCA dimensions, finding 100 to be the most effective.

4.2.3 Anomaly Detection Methods

A key advantage of the system presented by M2T is its unsupervised nature, and we seek to fulfil the same requirement by using only unsupervised anomaly detection methods. As stated earlier, unsupervised methods rely on 'normal' instances being far more frequent than their anomalous counterparts. With our analysis of the dataset in section 3.4 showing such low percentages of positive labels, we conclude that it is safe to make this assumption for our data. Following the logic of M2T, we explore the usage of a variety of methods, hoping to take advantage of the different properties of each method to create informative anomaly scores.

We first select methods that have previously been reported to perform well on high dimensional deep features:

- **K-Nearest Neighbour** (kNN) [12, 31] - The largest distance (Euclidean) from each sample point to its K nearest neighbours is considered and used as an anomaly score.
- **Multi-Variate Gaussian** (MCD) [38] - The data is modelled with a Multi-variate Gaussian Distribution, with a covariance matrix estimated with minimum covariance determinant (MCD) [39], and the Mahalanobis distance to each data point is used as its anomaly score.
- **One-Class Support Vector Machine** (OCSVM) [50] - A single class support vector machine is fitted to the data, with an radial basis function (rbf) kernel to learn a non-linear decision boundary. Outlier scores are based on distance to the fitted hyperplane.

In addition to previously used methods for anomaly detection on deep features, we also experiment with a variety of other methods used in M2T including:

- **AutoEncoder** - An autoencoder network is trained on the input data, and the reconstruction loss is used as the anomaly score.
- **Feature Bagging** - A meta estimator that fits a number of base detectors (in our case KNN) on sub-samples of the features of the dataset, and aggregates score across all detectors.
- **Local Outlier Factor** (LOF) - Similarly to KNN, for each sample K nearest neighbours are found. Instead of using the largest distance, however, the local density is computed. Samples with lower local densities are considered more anomalous.
- **Isolation Forest** - Each sample is isolated from every other sample through recursive axis-parallel subdivisions using a forest of decision trees. Those samples that are most easily isolated are determined to be the most anomalous.
- **Isolation-Based Nearest Neighbour Ensembles** (INNE) - An isolation-based approach similar to Isolation Forest, but uses a nearest neighbour based approach to perform isolation rather than an axis subdivision approach.

Methods employed by M2T such as Histogram Based Outlier Detection are disregarded due to their feature independence assumption, which cannot be made when dealing with deep features.

We utilise the PyOD (Python Outlier Detection) [53] library for anomaly detection, as it supports all selected methods.

4.2.4 Ensemble Method

Since different anomaly detection methods utilise different underlying logic, appropriately combining scores such that the combined score is more informative than any single score is a task with many possible solutions. In M2T, the authors use a simple majority voting rule to determine if a single frame should be anomalous. This works well when dealing with binary predictions, however in our case we are dealing with a continuous anomaly score outputted from each detector. Combining these scores is an area with relatively few published works, however Zhao et al [54] detail several options which we explore:

- **Mean Scoring** - Average all scores for each shot.
- **Top N Mean Scoring** - Average the top N scores for each shot.
- **Max Scoring** - Taking the maximum score for each shot.
- **Max Ranking** - Ranking each shot for each detector from least to most anomalous, and taking the maximum rank for each shot.
- **Mean Ranking** - Ranking each shot for each detector, and taking the mean rank for each shot.

Examining the chosen detector models, it becomes clear that each produces scores of different scales. Therefore, it is crucial to normalise the output of each detector [54] before using ensemble methods that directly utilise the score output such as Mean, Top N and Max scoring, such that no one detector dominates the ensemble score.

4.3 Results

Whether or not shots selected with this system will align with those used in a trailer is hard to determine without experimental results. The underlying assumption behind M2T is that how 'abnormal' a shot is, based on surface level audio and visual features, is a good indicator of the 'trailerness' of a shot, and this seems to result in good selections. However, the question of if the anomalies encoded by the deep features extracted for each shot will indicate suitability for a trailer is hard to determine. Considering a majority of film shots consist of fairly benign actions and image contents, hopefully the shots with anomalous content also prove to be ones suitable for a trailer. However, it may also be the case that the 'non-trailer' shots within a film are not sufficiently partitioned within the feature space from their trailer suitable counterparts to prove anomaly detection an effective method.

For our evaluation, we consider a budget of K=10 shots per trailer, creating trailers of roughly 2 minutes. Therefore, we take the top 10 shots with the highest anomaly scores and use them as the shot selection. Furthermore, we adjust the features used on a per shot basis to account for certain features missing from each shot. For example, for several shots there are no subtitles, resulting in a range of the extracted features missing. This is accounted for by simply passing the reduced set of feature anomaly scores to the ensemble method.

4.3.1 Quantitative Analysis

To analyse the performance of the system, we use accuracy as our metric, that is the percentage of correctly identified shots within the 10 selected. We first examine the performance of each individual anomaly detection method on the development set in Figure 4.2. We use all features and combine scores with a simple mean combination. We find that the methods used previously on deep features (OCSVM, MCD, KNN) all perform well. Using the intuition that the detectors which are unable to achieve better than random accuracy on shot selection will not be useful to the ensemble method, we disregard the scores from AutoEncoder, Isolation Forest and Feature Bagging in our future combination methods.

Next, we analyse the performance of different methods for combining the anomaly feature scores in Table 4.1. Again, we utilise all features, and find that using the top-N mean scoring method with N=3 gives the best results, with the max scoring method showing similar performance and the ranking methods showing poor performance. The hyperparameter N is tuned through a search from N=2 to N=10.

With the anomaly detection methods and combination method chosen, we perform an ablation study over the accuracy of the system using different features on both

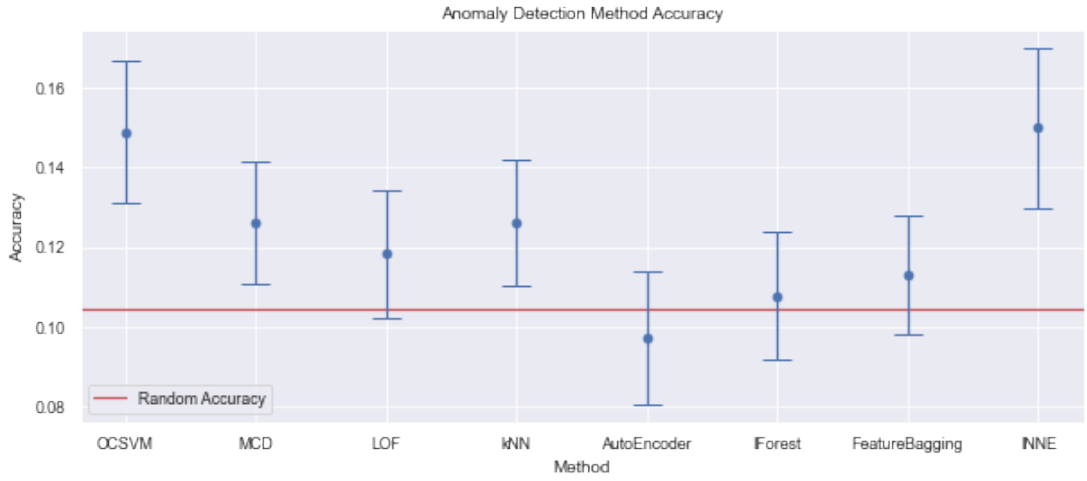

Figure 4.2: Anomaly Detection methods accuracy (dev) using mean combination across all extracted features. Abbreviations for anomaly detection methods are those used previously in Section 4.2.3.

Combination Method	**Accuracy (Dev)**
Random	0.103
Mean	0.155
Top-3 Mean	**0.171**
Max	0.168
Max Rank	0.116
Mean Rank	0.0895

Table 4.1: Accuracy results on development set using different combination methods.

the development and the test set in Table 4.2. The reported numbers are higher on the development set due to the development set including a range of trailers found for each film, increasing the number of positive labels per film, whereas the test set consists of labels from only the final official trailer per film. These results are promising, and show that a fusion of different features is able to produce better results than any one feature alone. We find that the frame and subtitles features show the strongest individual performance, with features extracted using CLIP proving more informative than those using ResNet. This can be attributed to the nature of the task each network is trained on: ResNet is trained purely for image classification, so features will lack an understanding of what is happening within an image, whereas CLIP is trained on image-text pairs, so produces features that encode an understanding of the content of a scene within an image. Whilst the results for some features are promising, we do observe the system failing to effectively take advantage of the audio and facial emotion features. An explanation for this is not clear, and we hope that future methods will be able to use these features more effectively to aid in shot selection, as intuitively both could be useful for the task.

The results are generally more mixed on the test set due to its noisier labels - if we assume that a film has many trailer suitable moments in it, positive labels for only a

Features	**Accuracy (Dev)**	**Accuracy (Test)**
Random	0.103	0.0418
Frame (CLIP)	0.147	0.0572
Frame (ResNet)	0.125	0.0502
Motion	0.126	0.0561
Face Emotion	0.108	0.0438
Audio	0.105	0.0488
Subtitles	0.113	0.0390
All Features	**0.171**	**0.0592**

Table 4.2: Ablation study examining the effectiveness of each feature in Anomaly Detection, Top-3 Mean Combination of all selected anomaly detection methods.

Group	**Genres**
Action	Action, Adventure, Fantasy, Sci-Fi, Western
Emotion	Comedy, Romance, Drama
Suspense	Thriller, Horror, Crime, Film-Noir, Mystery

Table 4.3: Genre grouping

single trailer will mean many false negatives, leading to noisier and less reliable metrics.

We further analyse the impact of film genre on the performance of the system. Film genres are obtained from IMDB [4]. To do this effectively, we first from three groups of similar genres shown in Table 4.3. These groups are based on manual inspection of trailers showing similar characteristics in type of shot used. This is done to increase the individual dataset sizes for more reliable metrics, since using the fine-grained genres results in datasets of only a couple of films per genre. We report the accuracy on each group in Table 4.4, and find that the system performs best on the Action and Suspense groups. This aligns with expectations; The genres present in these two groups generally have trailers characterised by the more 'bombastic' or action-heavy shots in a film, which we expect to be easier to detect through anomaly detection. The trailers for genres present in the Emotion group, however, have generally more nuanced trailers with shots which are less obviously different to the others in the film, and as a result suffer from worse accuracy scores.

4.3.2 Qualitative Analysis

Whilst the dataset contains soft labels for each shot based on similarity to trailer shots, these labels are not a comprehensive indicator of all the suitable trailer shots in a film, and the metrics presented may not paint the full picture of the effectiveness of the system. Therefore, we manually inspect the most and least anomalous shots from a variety of films to better understand the selections the system makes. [1]

[1]Trailers generated with shot suggestions from the Anomaly Detection system can be viewed here: `https://drive.google.com/drive/folders/1h7_6KXJnwrlwXPAfxZNhKC3xeJUOAJbI?usp=sharing`

Group	**Accuracy (Dev)**	**Accuracy (Test)**
Action	0.174	**0.0725**
Emotion	0.148	0.0497
Suspense	**0.213**	0.0647

Table 4.4: Results for different film genres, using all features

In Figure 4.3, we present key-frames from the most anomalous (top row) and least anomalous (bottom row) shots from a film in each of the genre groups. On manual inspection, we find that the most anomalous shots are especially effective as trailer shots in the 'Action' and 'Suspense' categories, with the generated trailers providing interesting moments and an overall effective trailer. This is in contrast to the accuracy metrics, for example with 'Total Recall' the generated trailer is quite effective, however it has a low accuracy of 0.1, indicating the accuracy metric is not an entirely reliable method for evaluating output quality. Whilst the most anomalous shots for the 'Action' and 'Suspense' groups are effective, the shots vary wildly in their content, with little consistency in a certain 'type' of shot being prevalent. This is to be expected given the nature of the system - it is not learning what makes an effective shot and therefore selected shots cannot be expected to be consistent in content. This inconsistency continues for the least anomalous shots, with several being particularly suitable for a trailer despite their low anomaly score, especially in the case of 'Total Recall' and 'The Shining'. Unsurprisingly, given the much lower accuracy score, the shot selections for films from the 'Emotion' group are even more inconsistent, with the generated trailers containing only few shots that could be construed as trailer worthy.

Overall, on manual inspection, the system shows mixed performance. In the range of shots identified as the most anomalous, many shots are shown to be suitable for a trailer and potentially 'anomalous', however this is not consistent across all of these shots. Furthermore, the system shows poor performance on the other end of the spectrum where it identifies 'mundane' shots but also mixes in shots that would not be out of place in a trailer. As expected, the system performance is vastly superior on films from the Action and Suspense groups, and we hope that more advanced methods will be better able to take advantage of the deep features to aid in shot selection for films from the 'Emotion' group.

Figure 4.3: Most and least anomalous shot selections for a film from each genre group

Chapter 5

Supervised Learning

5.1 Introduction

Whilst unsupervised approaches to trailer shot selection have shown significant success, newer models have benefited from directly using films alongside their trailers in a supervised manner. These methods disregard making direct assumptions about what makes an effective trailer shot and instead rely on training to identify these characteristics.

CCANet [49] accomplishes this with a novel architecture whereby soft labels are learnt whilst training, rather than using hard binary labels. They utilise Co-Attention between film shot features and all the corresponding trailer shot features to compute a soft label. This label is used within a ranking loss, rather than a binary cross entropy loss. The aim of this is to tackle the problem of labelling film shots that are similar to trailer shots with a hard label of 0 when using binary labelling, whilst they are still similar enough to warrant using as a positive training example. Furthermore, CCANet incorporates the usage of Contrastive Attention in order to augment the input features such that the trailer shot features stand out from features from non-trailer shots. Whilst these methods are shown to be effective, their usage is limited to only utilising visual features from both the trailer and the film, and the authors cite utilising more, multi-modal features as a possible avenue for future research.

Papalampidi et al [33] show that sacrificing these sophisticated mechanisms in favour of utilising more, multi-modal features alongside silver standard binary labels is in fact a more effective approach. Incorporating visual, audio and linguistic features, they train a transformer model on the binary task of identifying a shot as a trailer shot, obtaining results surpassing CCANet. Furthermore, scene-level features from the screenplay are utilised to identify Turning Points (TPs) and sentiment labels for each scene. These features are propagated to the shot-level and used to help inform trailer shot selection.

5.2 Baseline Model

We start with a baseline Multi-Layer Perceptron (MLP) model, trained on the binary classification task of identifying a shot as being in a trailer.

Hyperparameter	Values
Feature Projection Dimension	64, 128, 256
Number of Hidden Layers	1, 2, 3
Hidden Layer Dimension	64, 128, 256, 512

Table 5.1: MLP hyperparameter grid-search

Recognising that film genre is crucial to the type of shot selected for a trailer, we explore the use of multi-task learning to incorporate genre information into the training process. Here, we task the model with learning to predict both the 'trailerness' and the genre group of the input shot, as defined earlier in Table 4.3, whilst using a shared representation. The genre-groups are used, rather than fine-grained genres such as Comedy and Sci-Fi, such that there are sufficient training examples per class. Multi-Task learning is used in the hope that learning to predict the shot genre group helps the model learn a representation that will more effectively predict 'trailerness' in different genre groups.[1]

Architecture The model architecture is outlined in Figure 5.1. Each input feature (visual, linguistic, audio and emotion) is projected down to 128 dimensions with a linear layer, then all are concatenated to a single vector. This is then passed through two feed forward layers of 128 and 64 dimensions respectively with ReLU activation, before a final feed forward layer to 1 dimension with sigmoid activation for trailer prediction. These model hyperparameters are determined with a gridsearch over the feature projection dimension, number of hidden layers and hidden layer dimensionality, shown in Table 5.1. For the layer dimensions, we only test configurations where subsequent layers have a smaller dimension than the previous layer. When using multi-task learning, a second feed forward layer is used to project to 3 dimensions with softmax activation for genre prediction.

Implementation Details We implement our model in Tensorflow [1] with the Keras deep learning library [15]. We use the Adam optimiser [26] and train with a learning rate of $1 \cdot 10^{-4}$ for 20 epochs, with a dropout of 0.3. The high dropout is configured to avoid overfitting to the training data, which was occurring too quickly when using less dropout. The loss on both the training and development sets whilst training is shown in Figure 5.2, we can see the development loss for trailerness prediction plateaus after around 20 epochs and fails to improve further.

For the multi-task model, we combine binary cross-entropy loss for trailer prediction and categorical cross-entropy loss for genre prediction with a weighted sum, and find weights of $\alpha = 0.9$ and $\beta = 0.1$ to be the most effective.

$$Loss_{multitask} = \alpha \cdot Loss_{trailer} + \beta \cdot Loss_{genre}$$

[1] We also explore training individual models on the training data from each genre group, however find these to overfit too quickly before generalising well.

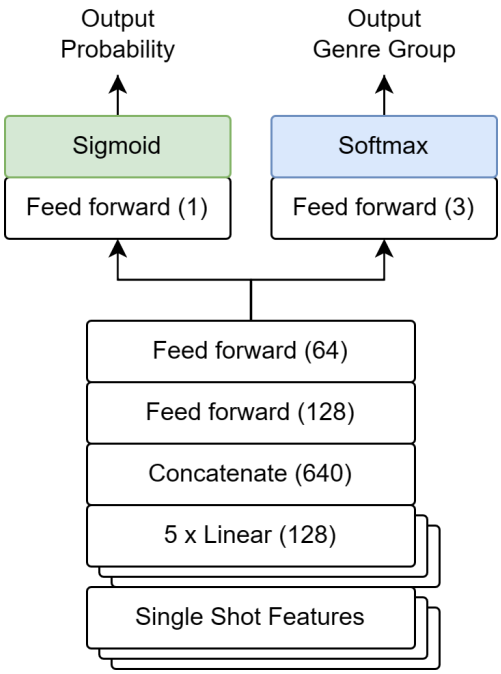

Figure 5.1: Architecture of baseline MLP model for multi-task learning, layer output shapes in brackets.

Features	**Accuracy (Dev)**
Frame (CLIP)	0.192
Frame (ResNet)	0.185
Motion	0.178
Faces	0.131
Audio	0.127
Subtitles	0.163
All Features	**0.211**

Table 5.2: Ablation study of the effectiveness of each feature for classification.

5.2.1 Results

We use the same accuracy metric presented in section 4.3.1, with each metric being averaged over 5 training runs with different seeds. First, we examine the performance of the system using each of the shot features in Table 5.2, and no genre information incorporated. We find that the baseline MLP model outperforms the anomaly detection in accuracy by a large margin. Again, CLIP features are shown to be the most valuable for the task, with ResNet features close behind. We also observe that the model is also able to effectively use the subtitle features, unlike the anomaly detection model.

The accuracies using different methods for incorporating genre information are shown in Table 5.3. We find that utilising multi-task learning is the most effective method for incorporating genre-specific information, and found that training genre-specific models quickly overfits the training data before it is able to generalize as well as the base model with no genre information. Films in the 'Emotion' genre group show the

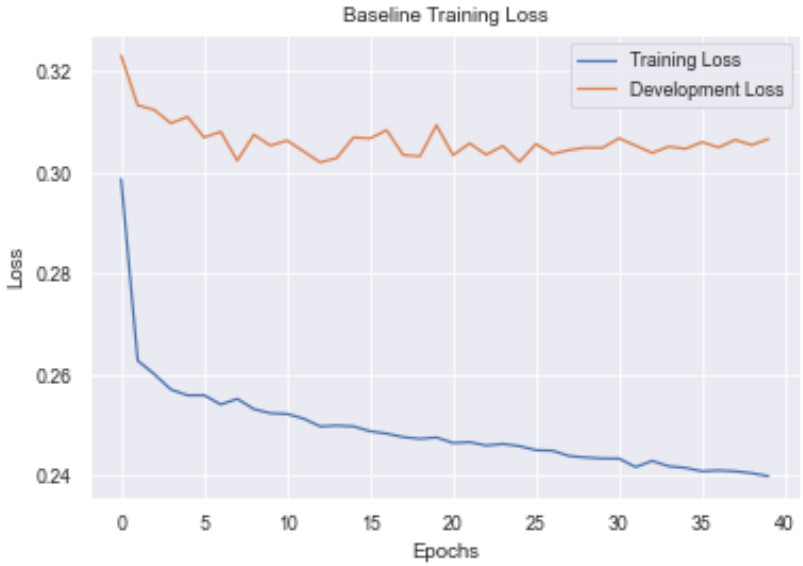

Figure 5.2: Loss on train and development sets whilst training baseline model.

MLP Model	Action	Emotion	Suspense
No Genre Information	0.218	0.206	0.212
Genre-Specific Model	0.203	0.198	0.205
Multi-Task Learning	**0.225**	**0.212**	**0.223**

Table 5.3: Development set accuracy after incorporation of genre information.

largest improvement over the anomaly detection model, increasing from 0.148 to 0.212. Whilst we cannot completely identify why this model is much better able to predict trailer shots for this group, the increase in accuracy in using subtitle features is a good indicator that the model is able to infer more subtle properties of shots, relevant for trailers from this group more than the others.

Examining the overall results in Table 5.4, we find that the baseline model with multi-task learning outperforms the anomaly detection model by a large margin. Whilst this is a good improvement, We will see that these results can be improved upon with a more sophisticated model.

5.3 Transformer Model

5.3.1 Motivation

Whilst the baseline model reports improvements over the anomaly detection method, there is still room for improvement. The baseline model is naive, in that it considers each film shot in isolation and judges its value as a trailer shot. Whilst this is effective, have we not established with the anomaly detection method that considering each shot **with respect to** the other shots within the same film is useful for effective shot recognition? Therefore, we can hypothesise that augmenting our baseline model to account for the

Model	**Accuracy (Dev)**	**Accuracy (Test)**
Random	0.103	0.0418
Anomaly Detection	0.171	0.0592
MLP + Multi-Task	**0.218**	**0.119**

Table 5.4: MLP Model comparison.

context of a shot, that is the other shots within the same film, will lead to improved performance.

This contextualisation of each input shot can be accomplished in a variety of ways, each with their own benefits and limitations. Recurrent models, such as an LSTM, can be used to produce contextualised feature vectors. Whilst using an RNN is a valid option, recurrent models suffer from a difficulty learning long distance dependencies [28], leading to the contextualised feature vector produced for each shot being weighted heavily towards those shots closest to it. This is not always an issue, however for our use case we predict that a representation with respect to the whole film will be much more informative than one with respect to the localised area around each shot. An alternative to the LSTM model that will allow this to be accomplished is the Transformer model [48]. This model has seen significant success in the field of natural language processing, due to its ability to learn arbitrary length dependencies and produce informative context vectors.

A key factor contributing to the success of a transformer is its method of contextualising input feature vectors with Multi-Head Attention. Using this, input feature vectors are contextualised with respect to every other input, with each 'head' in the multi-head attention module learning different relationships between inputs. Traditionally, Transformer encoders are stacked to produce deeper features, and whilst we will experiment with this, the large number of parameters each encoder layer introduces will likely make more than one encoder unrealistic with the limited amount of training data we have available.

Architecture The new model architecture is similar to the baseline model, but operates on a sequence of shot features rather than a single shot at a time. Each input feature (visual, linguistic, audio and emotion) is projected to 128 dimensions with a linear layer and concatenated as before, however after this the sequence of input features is fed through a transformer encoder with 4 attention heads and a 512 dimension intermediate representation. The contextualised output features are then projected to 1 dimension with a feed forward layer with sigmoid activation for trailerness prediction.

For multi-task learning, the contextualised feature vectors are average pooled and projected to 3 dimensions with a feed forward layer with softmax activation.

Model hyper-parameters such as number of attention heads and encoder intermediate dimension are determined through a grid-search over hyper-parameters configurations shown in Table 5.5. We also experiment with using positional encodings to incorporate location information for each shot, however we do not find them to improve performance.

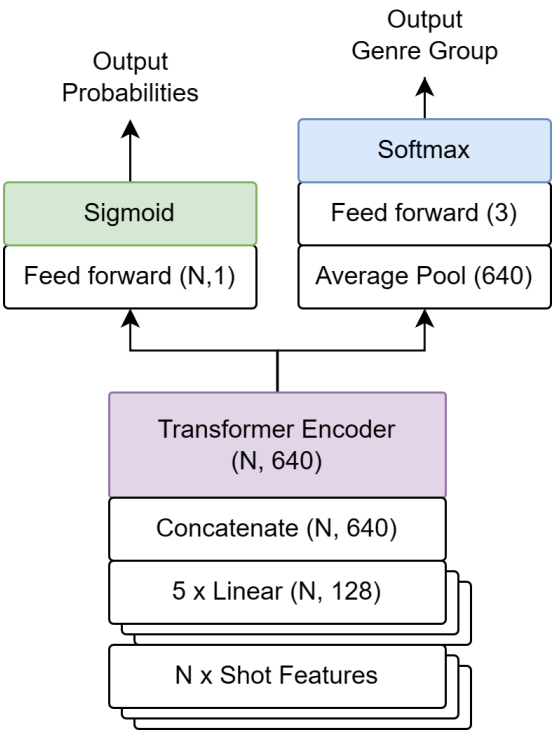

Figure 5.3: Architecture of Transformer Model, layer output shapes in brackets.

Hyperparameter	**Values**
Feature Projection Dimension	64, **128**
Intermediate Dimension	128, 256, **512**, 1024
Attention Heads	1, 2, **4**, 8
Encoders	**1**, 2

Table 5.5: Transformer hyperparameter grid-search.

Implementation Details Again, we use the Adam optimiser with a learning rate of $1 \cdot 10^{-4}$ and a dropout of 0.3. Our training, however, is not as straightforward as for the baseline model. Since our model is now accepting a sequence of shots rather than single shots, we must adjust our training to account for this.

We consider two training regimes for the transformer architecture:

1. Train with a batch size of one and input each film in its entirety. This strategy may be liable to overfitting and noisy training due to the single film batch size and the number of training examples being limited to the number of films.

2. For each sample in a batch, randomly sample positive and negative shots from a film. The motivation behind this is that there is always far more non-trailer shots than trailer shots, and using a random sampling method we can introduce a variety of contexts for training the transformer, and help prevent overfitting.

Experimenting with both, we find that using random sampling produces better results than using whole films as input. We posit that this is due to the variety of contexts

Figure 5.4: Loss on train and development sets whilst training transformer model.

Training Regime	**Accuracy (dev)**
Whole Film	0.241 (0.011)
Random Sampling	**0.252** (0.005)

Table 5.6: Training regime results, standard deviation in brackets. No genre information incorporated.

that arise from random sampling resulting in learning more generalized weights for the transformer encoder, leading to better generalization to the development and test sets.

Using the random selection method, we pick 9 random trailer shots and 21 random non-trailer shots from the same film for each sample and use a batch size of 16. We show the loss on the training and development in Figure 5.4, and see the loss plateau and begin to overfit around 60 epochs, where we early-stop.

5.3.2 Results

We first show the results from utilising each training regime described earlier in Table 5.6. We average results over 5 runs with different seeds. Not only do we observe that the random sampling method results in better performance, the standard deviation in accuracy is much lower than that from the whole film regime, confirming our assumption that using whole films as single inputs per batch would result in noisy training. Therefore, for all future experiments we utilise the random sampling training regime.

To confirm that the multi-task model is indeed jointly learning to predict trailerness and genre group, we examine the genre group prediction confusion matrix in Figure 5.5. We see that all classes are predicted with decent accuracy, thereby confirming that the multi-task learning is jointly learning genre-specific information whilst learning to predict trailerness. We do observe that the model is able to predict shots in films from

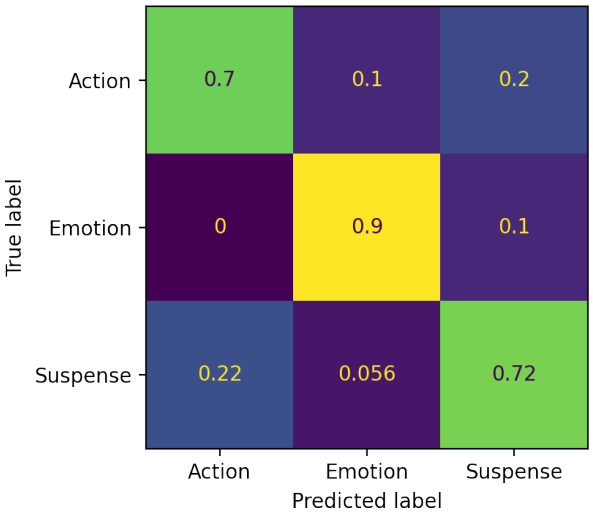

Figure 5.5: Genre Classification Confusion Matrix (Development set).

Model	**Action**	**Emotion**	**Suspense**
Anomaly Detection	0.174	0.148	0.213
MLP + Multi-Task	0.225	0.212	0.223
Transformer	0.255	0.249	0.256
Transformer + Multi-Task	**0.259**	**0.252**	**0.257**

Table 5.7: Accuracy on genre groups (Development set).

the 'Emotion' genre group with much more accuracy than the other classes. This may be due to the films this group being quite distinct from those in the Action and Suspense groups, whereas films in the other groups share several common characteristics. This hypothesis is further backed up by the observation that the Action and Suspense groups are most commonly misclassified as each other, whilst rarely being misclassified as the Emotion group.

Exploring the performance of the model for trailer shot detection on different genre groups in Table 5.7, we find that using multi-task learning boosts model performance only slightly across genre groups. Multi-task learning was found to improve results far more in the baseline MLP model, a result that can perhaps be explained by the nature of the Transformer model taking context into account and thereby not gaining as much information from the genre group target in comparison to the MLP model. Again, the model performs best on films from the Action and Suspense groups, however we find that the performance gap to the Emotion group is greatly reduced compared to the previous models.

We show the results from all the models in Table 5.8. The transformer model obtains the best results by a wide margin, confirming our hypothesis that utilising context for shot selection is essential to improve accuracy over considering each shot in isolation.

Model	Accuracy (Dev)	Accuracy (Test)
Random	0.103	0.0418
Anomaly Detection	0.171	0.0592
MLP + Multi-Task	0.218	0.119
Transformer + Multi-Task	**0.255**	**0.132**

Table 5.8: Comparison of accuracy from all models.

5.4 Qualitative Analysis

Whilst the transformer model reports higher accuracy on the development and test sets than the anomaly detection model, manual inspection of the selected shots is required to identify any trends or features that the model appears to be picking up on. Furthermore, as shown in section 4.3.2, the accuracy score is not a comprehensive indicator of the effectiveness of the predictions, making manual inspection essential.

Examining the trailers produced by the transformer model by hand[2], we observe a certain amount of consistency in the identified shots, unlike those outputted from the anomaly detection method. In Figure 5.6 we show the shots identified as most trailer-worthy and least trailer worthy from a film from each genre group. On inspection, we see that in these films and others the model is quite heavily biased towards shots featuring faces as the main element. In all three films shown in Figure 5.6, shots which feature a face close-up dominate both the most trailer-worthy shots shown in the diagram and those which aren't shown. Whilst consistently choosing the same style of shot is not ideal, we do find that the facial shots chosen by the model are 'emotionally charged' and therefore predominantly shots that would be suitable for a trailer. This observation is also helpful for explaining the observed increase in accuracy of shot selection for films from the Emotion group, since these shots are particularly suited to trailers for these films.

As for the shots that do not follow the full face theme, there is more variety and less of an obvious pattern. In each of the shots selected from the three groups, the wider angle shots often feature a more active subject and are well suited to a trailer. Furthermore, the model often selects shots featuring logos or title cards, with these featuring in the top shots for 500 Days of Summer and The Shining, and in particular featuring for almost **all** of the selected shots for the 'A Walk to Remember' trailer. This is unsurprising, since many trailers feature the logo and title cards and this would be apparent in the training data, however selecting these shots does not align with the goal of the model. The logo and title card shots can easily be added by an editor and our model recommending them is not particularly helpful. Removing these shots from the training data, therefore, could be a path to improving the shot recommendations from the model and we leave this to a future work.

All in all the model performs well, however it is by no means perfect. Each generated trailer features shots that are questionable in the context of a trailer or preview, and

[2]Trailers generated with shot suggestions from the Transformer model can be viewed here: `https://drive.google.com/drive/folders/1gt8NS6lsGDRQUWIr5S2VH62ZybG0pGrY?usp=share_link`

Figure 5.6: Transformer model shot selection

combined with the black-box nature of the model this makes for a system that is tricky to debug. Furthermore, whilst the accuracy metrics for this model are vastly superior to those from the Anomaly Detection model, on manual inspection several of the Anomaly Detection trailers are shown to be equally, if not more, interesting than their supervised counterparts. For example, in the '2012' trailer the Anomaly Detection model selects many interesting, action-heavy shots, whereas the supervised model shows a much more mixed performance, with a couple of interesting shots but also many of the previously discussed face close-ups. This could be attributed to the nature of the supervised model - it predicts shots similar to those it has seen in its training data, leading to the inclusion of many title cards and face close-ups, but less consistency in the other shots chosen. Despite the Action and Suspense genres scoring the best accuracy metrics, the Emotion films appear to be the most improved over their Anomaly Detection counterparts, with the dialogue scenes chosen consistently being more interesting and 'emotionally heavy' than those chosen by the Anomaly Detection model. We posit that this is an indicator that the model is successful in the original goal of utilising deep features to choose shots based on a deeper understanding of the content of a shot.

Chapter 6

Conclusions

In this project we explored the task of automated film shot selection for trailers. We utilised previous work to hypothesise that multi-modal, deep features extracted with modern deep neural networks would provide the necessary information to effectively develop methods for shot selection. Using the previously introduced TRIPOD$\bigoplus$ dataset [33], we extracted a variety of multi-modal features with state-of-the-art deep learning models. We then presented two methods using these features; First, an unsupervised system based on Anomaly Detection taking inspiration from movie2trailer [46], and second, a supervised method utilising the Transformer architecture for shot labelling using noisy training labels (obtained through similarity scores between trailer and film shot features).

Our evaluation of the Anomaly Detection system finds that previously reported methods for anomaly detection on deep features are also helpful for our use-case. Furthermore, we find the system is capable of producing effective shot selections for the films from genres such as Action and Horror, however we observe it failing to produce consistent results for films requiring more nuanced selections in genres such as Romance or Drama. On manual inspection of generated shot selections, we find that the accuracy metric used for quantitative analysis is not a completely reliable indicator of the effectiveness of the selected shots, with film trailers showcasing effective shot selections achieving lower than expected accuracy scores.

Turning to supervised methods, we experiment with both a simple MLP model and a Transformer model, finding the Transformer to obtain superior accuracy scores by a large margin, and we utilise multi-task learning to further augment model performance. On manual inspection of the shots selected, we observe much more consistency in the style of shot chosen than with the Anomaly Detection model, a promising sign that the supervised system is indeed utilising training data effectively to identify characteristics of effective shots. However, we find that this comes at the cost of more repetitive shot selections than those observed by the Anomaly Detection system, with face close-ups and titles cards dominating the selected shots. This consistency certainly helps the model to achieve superior and more consistent accuracy scores, however we find that trailers produced by the Anomaly Detection system are sometimes superior to those from the supervised model, even when reporting a lower accuracy metric.

Given these observations, we conclude that utilising human evaluation of selected shots is essential to effectively quantify the quality of the output from each method. Whilst the presented accuracy metric is useful for development, its unreliable nature opens avenues for future work, such as incorporating better automatic metrics and large-scale human evaluation.

In terms of future work, we see several opportunities to improve upon the supervised method presented. Firstly, manually cleaning the training data of logo and title-card shots should help the model to avoid predicting these as appropriate for a trailer. Furthermore, whilst we attempt to incorporate emotion information from faces, we foresee that utilising datasets and models dedicated to multi-modal emotion detection in video would provide even more informative features than those already used. A possible avenue for this could be to utilise the LIRIS-ACCEDE [11] dataset alongside a model such as MMDQEN [55], a dataset and model dedicated to *affective content analysis*, to extract more informative emotion-based features. Another limitation of our model is our reliance on the noisy binary labels obtained through soft labelling. As observed by the authors of CCANet [49], this strategy leads to negative labels on shots that are very similar to trailer shots, and as such increasing the noisy nature of the training data. To combat this, we could use strategies such as the Co-Attention mechanism proposed by CCANet to learn soft labels whilst training, rather than relying on hard binary labels. This strategy, combined with the more informative multi-modal features extracted not used in CCANet, is a promising avenue for future research. Finally, as explored in our Introduction, automatically generated content tailored to viewers is a topic that is seeing a surge of interest. As such, developing techniques for conditioning the supervised methods on viewer specific requirements is an interesting direction for future research. However, the resources required in obtaining the data necessary for exploring such an avenue may limit its development to private companies.

Bibliography

[1] TensorFlow: Large-scale machine learning on heterogeneous systems. https://www.tensorflow.org/, 2015. Software available from tensorflow.org.

[2] Yamnet. https://github.com/tensorflow/models/tree/master/research/audioset/yamnet, 2020. [Online; accessed 20-02-2023].

[3] How many shots are in the average movie? https://stephenfollows.com/many-shots-average-movie/, 2023. [Online; accessed 04-04-2023].

[4] Imdb: Ratings, reviews, and where to watch the best movies. https://www.imdb.com/, 2023. [Online; accessed 20-02-2023].

[5] Netflix statistics & facts that define the company's dominance in 2023. https://www.comparitech.com/blog/vpn-privacy/netflix-statistics-facts-figures/, 2023. [Online; accessed 04-04-2023].

[6] Netflix tests pre-roll video 'previews' that are personalized to your interests. https://techcrunch.com/2017/06/19/netflix-tests-pre-roll-video-previews-that-are-personalized-to-your-intere 2023. [Online; accessed 04-04-2023].

[7] Pyscenedetect: Video scene cut detection and analysis tool. https://github.com/Breakthrough/PySceneDetect, 2023. [Online; accessed 20-02-2023].

[8] Tiktok statistics. https://techjury.net/blog/tiktok-statistics/, 2023. [Online; accessed 04-04-2023].

[9] Youtube: Marvel studios' secret invasion - official trailer. https://www.youtube.com/watch?v=Tp_YZNqNBhw&ab_channel=MarvelEntertainment, 2023. [Online; accessed 04-04-2023].

[10] Mohiuddin Ahmed, Abdun Naser Mahmood, and Md. Rafiqul Islam. A survey of anomaly detection techniques in financial domain. *Future Generation Computer Systems*, 55:278–288, 2016.

[11] Yoann Baveye, Emmanuel Dellandréa, Christel Chamaret, and Liming Chen. Liris-accede: A video database for affective content analysis. *IEEE Transactions on Affective Computing*, 6(1):43–55, 2015.

[12] Liron Bergman, Niv Cohen, and Yedid Hoshen. Deep nearest neighbor anomaly detection, 2020.

[13] Alexey Bochkovskiy, Chien-Yao Wang, and Hong-Yuan Mark Liao. Yolov4: Optimal speed and accuracy of object detection. 2020.

[14] Varun Chandola, Arindam Banerjee, and Vipin Kumar. Anomaly detection: A survey. *ACM Comput. Surv.*, 41(3), jul 2009.

[15] François Chollet et al. Keras. `https://keras.io`, 2015.

[16] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009.

[17] Jiankang Deng, Jia Guo, Yuxiang Zhou, Jinke Yu, Irene Kotsia, and Stefanos Zafeiriou. Retinaface: Single-stage dense face localisation in the wild, 2019.

[18] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding, 2018.

[19] C. Feichtenhofer, H. Fan, J. Malik, and K. He. Slowfast networks for video recognition. In *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 6201–6210, Los Alamitos, CA, USA, nov 2019. IEEE Computer Society.

[20] Bhagyashree Gaikwad, Ankita Sontakke, Manasi Patwardhan, Niranjan Pedanekar, and Shirish Karande. Plots to previews: Towards automatic movie preview retrieval using publicly available meta-data. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV) Workshops*, pages 3205–3214, October 2021.

[21] Jort F. Gemmeke, Daniel P. W. Ellis, Dylan Freedman, Aren Jansen, Wade Lawrence, R. Channing Moore, Manoj Plakal, and Marvin Ritter. Audio set: An ontology and human-labeled dataset for audio events. In *Proc. IEEE ICASSP 2017*, New Orleans, LA, 2017.

[22] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778, 2016.

[23] Minyoung Huh, Pulkit Agrawal, and Alexei Efros. What makes imagenet good for transfer learning? 08 2016.

[24] Go Irie, Takashi Satou, Akira Kojima, Toshihiko Yamasaki, and Kiyoharu Aizawa. Automatic trailer generation. In *Proceedings of the 18th ACM International Conference on Multimedia*, MM '10, page 839–842, New York, NY, USA, 2010. Association for Computing Machinery.

[25] Shijoe Jose, D. Malathi, Bharath Reddy, and Dorathi Jayaseeli. A survey on anomaly based host intrusion detection system. *Journal of Physics: Conference Series*, 1000(1):012049, apr 2018.

[26] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization, 2014.

[27] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton. Imagenet classification with deep convolutional neural networks. *Commun. ACM*, 60(6):84–90, may 2017.

[28] Phong Le and Willem Zuidema. Quantifying the vanishing gradient and long distance dependency problem in recursive neural networks and recursive lstms, 2016.

[29] Chun-Liang Li, Kihyuk Sohn, Jinsung Yoon, and Tomas Pfister. Cutpaste: Self-supervised learning for anomaly detection and localization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9664–9674, June 2021.

[30] Linjie Li, Yen-Chun Chen, Yu Cheng, Zhe Gan, Licheng Yu, and Jingjing Liu. Hero: Hierarchical encoder for video+ language omni-representation pre-training. In *EMNLP*, 2020.

[31] Tiago Santana Nazaré, Rodrigo Fernandes de Mello, and Moacir Antonelli Ponti. Are pre-trained cnns good feature extractors for anomaly detection in surveillance videos? *ArXiv*, abs/1811.08495, 2018.

[32] Pinelopi Papalampidi, Frank Keller, and Mirella Lapata. Movie plot analysis via turning point identification, 2019.

[33] Pinelopi Papalampidi, Frank Keller, and Mirella Lapata. Film trailer generation via task decomposition. 11 2021.

[34] Matthew E. Peters, Sebastian Ruder, and Noah A. Smith. To tune or not to tune? adapting pretrained representations to diverse tasks, 2019.

[35] Matthew E. Peters, Sebastian Ruder, and Noah A. Smith. To tune or not to tune? adapting pretrained representations to diverse tasks, 2019.

[36] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision, 2021.

[37] Nils Reimers and Iryna Gurevych. Sentence-bert: Sentence embeddings using siamese bert-networks, 2019.

[38] Oliver Rippel, Patrick Mertens, and Dorit Merhof. Modeling the distribution of normal data in pre-trained deep features for anomaly detection, 2020.

[39] Peter J. Rousseeuw and Katrien Van Driessen. A fast algorithm for the minimum covariance determinant estimator. *Technometrics*, 41(3):212–223, 1999.

[40] Andrey V. Savchenko. Facial expression and attributes recognition based on multi-task learning of lightweight neural networks. In *Proceedings of the 19th International Symposium on Intelligent Systems and Informatics (SISY)*, pages 119–124. IEEE, 2021.

[41] Andrey V. Savchenko. Video-based frame-level facial analysis of affective behavior on mobile devices using efficientnets. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, pages 2359–2366, June 2022.

[42] Andrey V Savchenko, Lyudmila V Savchenko, and Ilya Makarov. Classifying emotions and engagement in online learning based on a single facial expression recognition neural network. *IEEE Transactions on Affective Computing*, 2022.

[43] Sheng Shen, Liunian Harold Li, Hao Tan, Mohit Bansal, Anna Rohrbach, Kai-Wei Chang, Zhewei Yao, and Kurt Keutzer. How much can clip benefit vision-and-language tasks?, 2021.

[44] Alan F. Smeaton, Bart Lehane, Noel E. O'Connor, Conor Brady, and Gary Craig. Automatically selecting shots for action movie trailers. In *Proceedings of the 8th ACM International Workshop on Multimedia Information Retrieval*, MIR '06, page 231–238, New York, NY, USA, 2006. Association for Computing Machinery.

[45] John Smith, Dhiraj Joshi, Benoit Huet, Winston Hsu, and Jozef Cota. Harnessing a.i. for augmenting creativity: Application to movie trailer creation. pages 1799–1808, 10 2017.

[46] Domen Tabernik, Alan Lukezic, and Klemen Grm. movie2trailer: Unsupervised trailer generation using anomaly detection.

[47] Du Tran, Lubomir D. Bourdev, Rob Fergus, Lorenzo Torresani, and Manohar Paluri. C3D: generic features for video analysis. *CoRR*, abs/1412.0767, 2014.

[48] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. Attention is all you need. In I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017.

[49] Lezi Wang, Dong Liu, Rohit Puri, and Dimitris N. Metaxas. Learning trailer moments in full-length movies with co-contrastive attention. In Andrea Vedaldi, Horst Bischof, Thomas Brox, and Jan-Michael Frahm, editors, *Computer Vision – ECCV 2020*, pages 300–316, Cham, 2020. Springer International Publishing.

[50] Yinan Wang, Ryota Yoshihashi, Kawakami Rei, Shaodi You, Tohru Harano, Masahiko Ito, Katsura Komagome, Makoto Iida, and Takeshi Naemura. Unsupervised anomaly detection with compact deep features for wind turbine blade images taken by a drone. *IPSJ Transactions on Computer Vision and Applications*, 11, 12 2019.

[51] Hongteng Xu, Yi Zhen, and Hongyuan Zha. Trailer generation via a point process-based visual attractiveness model. In *Proceedings of the 24th International Conference on Artificial Intelligence*, IJCAI'15, page 2198–2204. AAAI Press, 2015.

[52] Beibei Zhang, Fan Yu, Yanxin Gao, Tongwei Ren, and Gangshan Wu. Joint learning for relationship and interaction analysis in video with multimodal feature

fusion. In *Proceedings of the 29th ACM International Conference on Multimedia*, MM '21, page 4848–4852, New York, NY, USA, 2021. Association for Computing Machinery.

[53] Yue Zhao, Zain Nasrullah, and Zheng Li. Pyod: A python toolbox for scalable outlier detection. *Journal of Machine Learning Research*, 20(96):1–7, 2019.

[54] Zhiruo Zhao. *Ensemble Methods for Anomaly Detection*. PhD thesis, 2017. Copyright - Database copyright ProQuest LLC; ProQuest does not claim copyright in the individual underlying works; Last updated - 2022-10-31.

[55] Yaochen Zhu and Zhenzhong Chen. Affective video content analysis via multimodal deep quality embedding network. *IEEE Transactions on Affective Computing*, PP:1–1, 06 2020.

[56] Arthur Zimek, Erich Schubert, and Hans-Peter Kriegel. A survey on unsupervised outlier detection in high-dimensional numerical data. *Statistical Analysis and Data Mining: The ASA Data Science Journal*, 5(5):363–387, 2012.