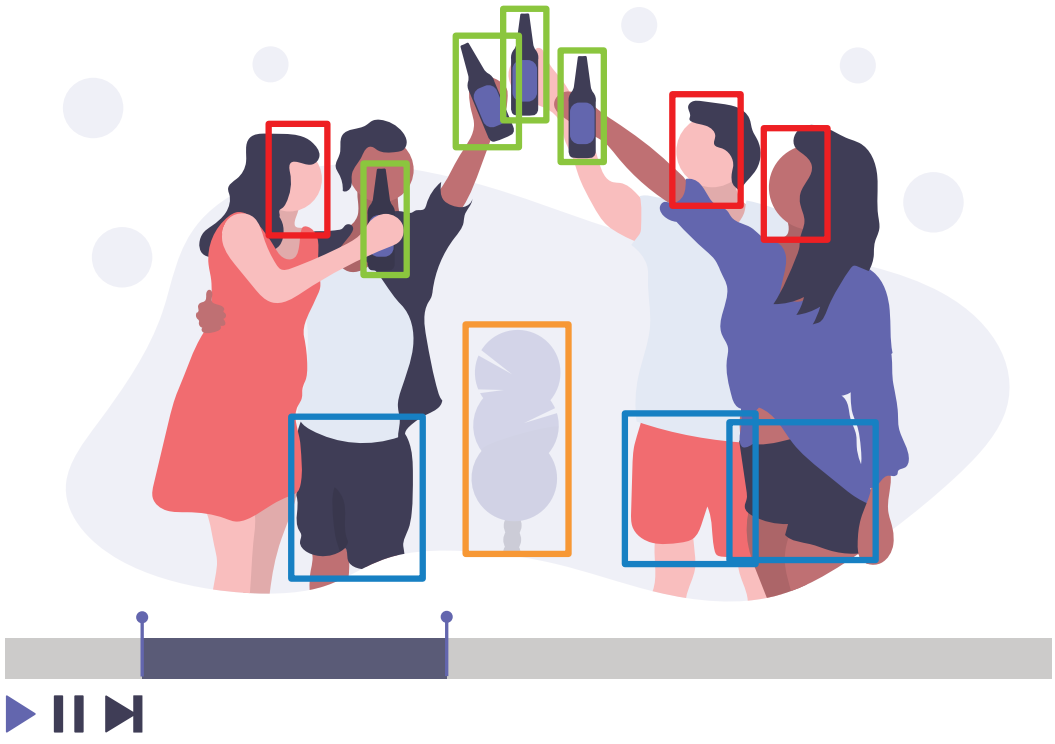
UNIVERSITY OF TWENTE.

BEELD EN GELUID

A PICTURE IS WORTH A 1000 WORDS

Introducing the visual modality to the query dependent video clip selection process



Janwillem te Voortwis
S1441620
EEMCS - Interaction Technology
January 2021

Supervisors:
Gwenn Englebienne *UTwente*
Mannes Poel *UTwente*
Roeland Ordelman *BenG*

Abstract

Video search and recommendation systems that work with long videos that cover multiple subjects have a unique feature. These videos often contain only parts that are relevant to the user. Providing access to the relevant parts in the video quickly and easily is favourable for the user, this can be done using query dependent clip selection. The visual relevance and visual interest of these clips is important when only the visual modality is shown to users. The perceived system performance by the users might not be satisfactory when the relevant information only resides in other modalities, the lacking visual information does not satisfy the user.

We propose a system that includes the visual modality in the clip selection process to improve the visual relevance and visual interest. The information from the speech and visual modalities are transformed to the same semantic feature space to be able to compare clips to the query in this same feature space. The proposed system is evaluated on visual relevance and visual interest against a comparable system that does not use the visual modality.

Including the visual modality does significantly improve the visual relevance of the clip selection. The visual interest of the selections improved significantly, but not as convincingly as visual relevance. This research suggests that the type of query has an influence on the performance of the proposed system.

Table of Contents

Chapter 1

Introduction

Due to the explosive growth and widespread availability of video nowadays, and the effort taken to process this content, new ways need to be found to efficiently and effectively offer the right content to the right users. This is often done through video search or recommendation systems to supply the user with videos that are relevant to them. These systems often work on a document level, meaning that the systems return complete videos. This works well for entertainment systems like Netflix or YouTube, but it becomes a bit more tedious when working with, for example, long videos which cover multiple subjects and contain only parts that are relevant to the user. For example, a user is searching for singing birds. If the videos are returned as whole documents, the user needs to watch the whole video to find that one clip with the singing birds. The same holds for a recommendation system, it can be that only a part of the video is interesting and relevant for the user. And even when the whole document is relevant or interesting, it is favourable to not having to watch the whole video to decide that you want to watch it.

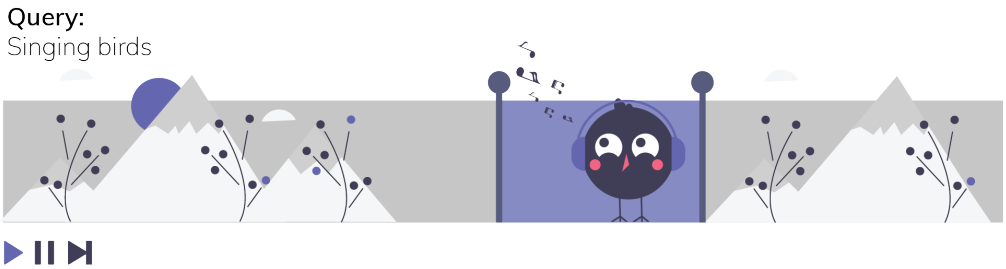


Figure 1.1: Select a relevant part of a video dependent on a query

The possibility that only a part of a video is relevant or interesting, is a problem that the Netherlands Institute for Sound and Vision (NISV) faces when offering their content to users. NISV collects, looks after, and provides access to the Dutch audio-visual heritage. In total, the collection holds more than a million hours of television,

radio, music and film that began in 1898 and continues to grow daily. Almost all the documents in the archive are of broadcast length (20 min or longer) and a big part of the archive consists of news broadcasts. These news broadcasts cover multiple subjects and together with the long length of all the documents in this archive, the need for a system that selects interesting and relevant clips becomes apparent.

To shorten the time the user must spend on each video, video summarization [1] and highlight detection [2] are often used to give a short preview of the full video. This way the user does not have to watch the full video to see if it contains interesting or relevant parts. Video summarization produces a summary of a full-length video and highlight detection is used to select parts of the video that are interesting to the user. These systems can be used separately or together, but they do not use any input given by a search or recommender system. This means that the systems might select parts of the video that are not relevant or interesting to the users given input, while omitting parts that are. When searching for a segment from a news broadcast, the summarization system would still summarize the whole broadcast, not the parts relevant to the user. And the highlight detection might detect highlights from the broadcast that have nothing to do with the sought-for segment. To get segments that better match the user's query, the selection systems should be query dependent.

Figure 1.2: Select a relevant to the query part of a video using the audio transcript

The NISV introduced a query-dependent video clip selection system that selects a clip from videos that best suits the query. By segmenting the visual documents into small pieces, efficient access is provided to the relevant content [3] [4]. Automatic

Speech Recognition (ASR) transcripts and subtitle data from the videos are used to find the clip that is most relevant for the query as illustrated in figure 1.2.

An observation made by the NISV is that these selected clips often lack visual relevance. For example, a query contains keywords about pandas but the selected videoclip contains no images of pandas. It could be that someone is talking about pandas in the clip, but they are never shown. The clip is relevant, but not inherently visually relevant. Another observation made by the NISV is that the selected clips are not very visually interesting. For example, news anchors who talk about items that correlate with the query behind their desks might not be perceived as visually interesting.

Video clips not being visually interesting and lacking visual relevancy becomes a problem when only the visual modality is shown to users. The perceived system performance by the users might not be satisfactory. The user expects relevant and interesting clips, but when the relevant information from the audio is not given, the lacking visual information will not satisfy the user. The system might work well and select clips which are relevant or interesting based on the audio or metadata, but the perceived performance by the user using the visual modality does not reflect that same relevance or interest.

Figure 1.3: Perceived performance of the user using the visual modality does not reflect the relevance or interest of the speech/audio modality

The lack of visual relevance and clips not being visually interesting might come from using non-visual information, like ASR and subtitle data, to rank visual content. The visual modality is ignored in the video clip selection process and might not achieve visually satisfactory results [5]. It is hypothesised that including the visual modality in the clip selection process improves the visual relevancy of the selections and makes the selections more visually interesting.

1.1 Research questions and challenges

These problems and challenges translate in the following research question:

> *How does including the visual modality in a query dependent clip selection process affect the visual relevance of the selections and how visually interesting these selections are?*

To answer this question some challenges need to be overcome. First, including the visual modality in the clip selection process. The visual information needs to be extracted from the videos through visual features. These visual features need to augment the video clip selection process. Second, the visual modality cannot be directly compared to a textual query. A translation step needs to be made from the visual domain to the semantic domain. Third, the temporal dimension of the video plays a role in the clip selection process. The best start and stop boundaries of the clips need to be found. The videos need to be segmented in a way so that coherent clips can be selected which then can be ranked based on how well they match the query. Temporal data in this context is defined as data of which is known at what timestamp in the video it was recorded. For example, subtitle data tells what is said in the video, but more importantly, when it is said.

To answer the research question, a multi-modal query-dependent video clip selection system (VCSS) is proposed that can return video clips which are visually relevant and interesting for the user. This selection system includes the visual and speech modality to select clips that are relevant to the query. The aforementioned challenges are overcome by building a prototype of this VCSS. Finally, this VCSS prototype is evaluated against an existing system.

The VCSS prototype mimics the tasks of an earlier prototype system developed by the NISV. To avoid confusion the prototype system developed by the NISV will be referred to as the NISV system. This NISV system is described in section 1.3. Both the VCSS prototype and the NISV system retrieve video clips from the NISV archive that are relevant and interesting to a written news article. The NISV system is used as a baseline in the evaluation of the VCSS on visual relevance and visual interest of the selected video clips.

1.2 Dataset

The dataset used for both clip selection systems is the archive collection of the NSIV. For this research, the television segment[1] of this collection is used. Almost the entirety of this segment consists of public service TV. There are several different TV programs, but the majority (one third) consists of news broadcasts.

1.3 NISV system (baseline)

The Netherlands Institute for Sound and Vision (NISV), Beeld en Geluid in Dutch, has developed a video wall (called News behind the News) that shows videoclips from their video archives related to current news articles. Each time a news article is published, the system shows 6 videoclips from the archives related to this news article. The video wall interface can be seen in figure 1.4. Currently, the system gets news articles by listening to multiple RSS feeds. The text from these articles is sent to a search API, which in turn returns the top six most relevant clips from the video archive. These clips are then cashed in an application server so the visitors can view these clips on the museum interface. This process is illustrated in figure 1.5.

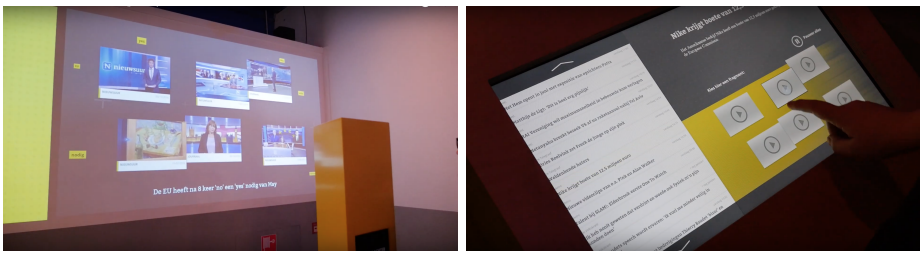


Figure 1.4: Interface of the News behind the news system

The input for this system is not a descriptive query. A descriptive query would be keywords or a description of what you want to find, the NBTN application uses a news article as a query. The article contains keywords, but around those keywords is a lot of context and some irrelevant text. These articles need to be processed to extract relevant information to be able to search the video archive. The NISV system does this by removing commonly used words (stop words) and counting word occurrences in the news article. This information is then used to select the most relevant videos.

The NISV system does not select a clip from the video, it presents a jump-in point. The system selects the best place to start playing the video but does not

[1] `https://archiefstats.beeldengeluid.nl/flight-over-the-archive/tv`

tell where the relevant clip ends. These jump-in points are determined in three ways. Which method is used depends on the data that is available for the video that is being processed. The first method uses annotated sections. These manually annotated sections divide a television program in coherent parts each with their own description. For example, a news broadcast is divided into news items. Each section would describe a news item. The start of the most relevant section can then be used as a jump-in point. Only a small part of the archive is annotated with sections.

The other two methods use a transcription of spoken text, either subtitles or ASR. The temporal data of each sentence in the transcript is used for clip selection. The most relevant sentence is found by comparing the keywords to the content of the sentences. Then the start time of this sentence is used as the jump-in point for the video.

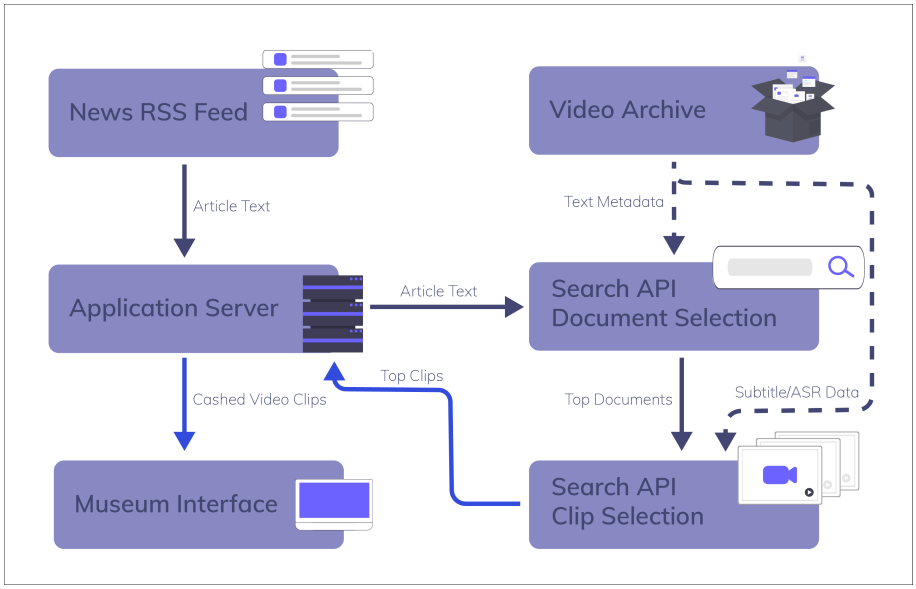


Figure 1.5: Current process of video clip selection for the ”News behind the News” installation

Chapter 2

Related Work

Although video retrieval, recommendation and highlight selection are becoming quite mature fields of research, query-dependent video clip selection is a relatively uncharted territory. But there are multiple related fields that can help understand the challenges that might present themselves.

2.1 Query dependent clip selection systems

As early as 1995, query dependent clip selection showed potential in the form of the Informedia project [3] [6] [7]. The Informedia project established a large, online digital video library featuring full-content and knowledge-based search and retrieval. The first publication by A.G. Hauptmann and M. A. Smith [3] proposed a system for retrieving a short video paragraph in response to the user's query. This system searches for keywords from the user query in the speech transcripts of the videos. A video paragraph that surrounds the best matching part of the transcript is returned as the most relevant clip. They define a video paragraph as a part of the video that starts at a natural boundary of the relevant content and ends wherever the video moves to a different context. Video paragraphs are created by matching audio paragraphs to the nearest shot transition. This process is illustrated in figure 2.1. Audio paragraphs are created by looking for breaks with silence in the audio track.

Later publications of the Informedia project [6] [7] do not mention the query dependent clip selection anymore. Instead they are using video skimming, which is a video summarization technique.

A fairly similar method to that of the first Informedia publication, is one proposed by Tat-Seng and Li-Qun [4]. Each video shot is logged using text descriptions, audio dialogue, and cinematic attributes. A two-layered, concept-based model is used. The first layer contains the video shots which are linked to a second layer contain-

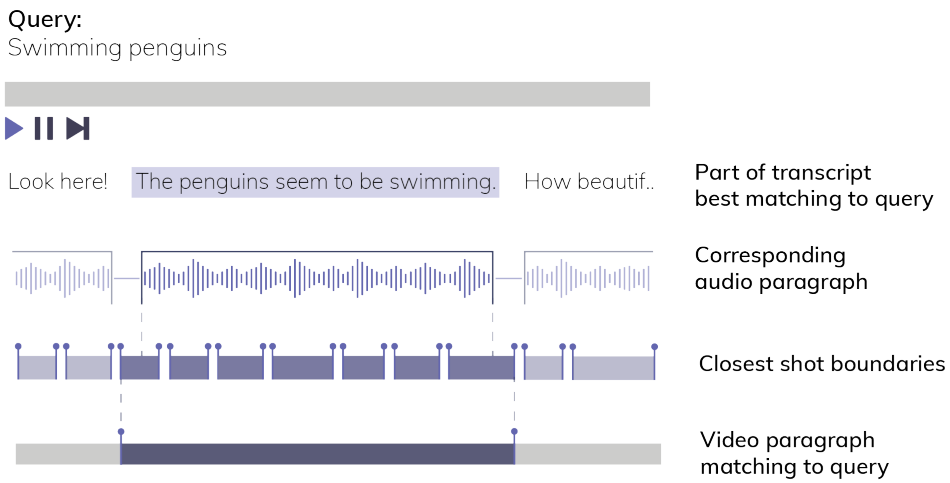


Figure 2.1: Selecting video paragraph that surrounds the best matching part of the transcript.

ing scenes. The scene layer is used to describe the overlapping concepts of the underlying shots. The information from both layers is used for accurately retrieving relevant video shots based on users' free-text queries. This method and the Informedia method heavily relied on manual shot and scene annotation. They had systems in place to suggest shot and scene boundaries, but in the end these suggestions were manually curated.

2.2 Introducing the visual modality

The previous mentioned methods used the visual modality only for the segmentation process, not as information for the selection of the best video segment. To introduce the visual modality in the selection process, features need to be extracted from the video. There are two types of features, low level and high level features. Low level features is knowledge extracted from the document data without a classification or detection step. This is a direct translation of the data. High level features are outputs of detectors and classifiers that use the document as an input. To be meaningful to the clip selection process these extracted visual features need to contain contextual information. These features will be used to match clips to a textual query. Since low level features only offer a translation of the data and do not extract contextual information, they will not be useful to the clip selection process. This leaves the high level features.

The most commonly used visual high level features are concepts [8]. Typically,

concepts are objects, people, places, events and actions. Visual concepts are extracted from a video using concept detectors. For example, using You Only Look Once (YOLO) object detection [9]. An output of such a concept detector can be found in figure 2.2.

The list of possible recognisable concepts is predetermined and relations between these concepts are often included in this list. For example, the concepts apple and pear both have a relation to the concept of fruit. The concept list with their relations form an ontology that can be utilised when matching concepts to the query. This way an exact match between concepts in the query and the document is not needed. For example, when the concept apple is found in the query and the concept pear is detected in the document, then the document will still be regarded as a close match to the query without being an exact concept match. Some systems exploit this relationship, for example, Sang et al. [10] use the conceptual relationships to find semantic similarity between documents. A typical system uses concept detectors both on the query and the documents to extract the concepts. Then the found concepts in the documents can be matched to the concepts in the query, leveraging the relations in the ontology of the concepts.

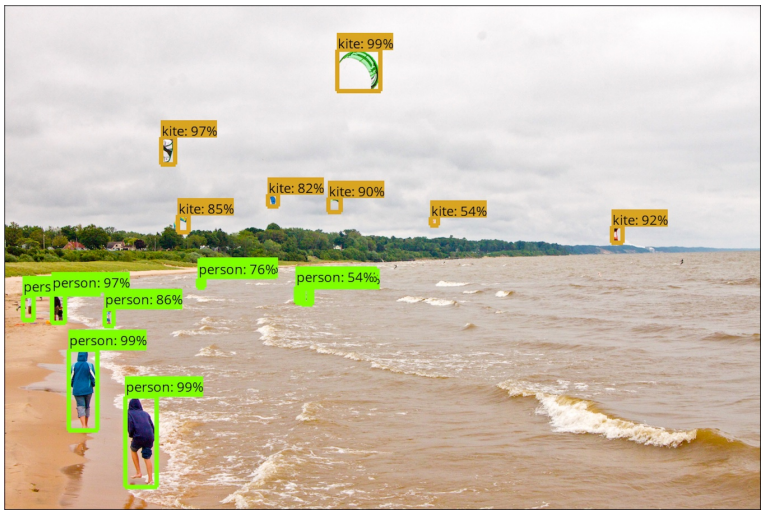


Figure 2.2: Output of an object detection system

2.3 Translation from visual domain to semantic domain

A problem with introducing visual features in the clip selection process is that these visual features cannot be directly used to compare to a textual query. A translation step needs to be made from the visual domain to the semantic domain.

A method by Natsev et al. [11] attempt to make this translation step. They propose a method that uses visual concepts to do semantic query expansion. Visual concepts are extracted from the videos using concept detectors. These concepts are then utilised to expand the original query to get a new set of results. Two methods are proposed to find words to expand the query with. The first method takes a lexical approach. This approach leverages global language properties, such as synonyms. Each concept has a short description that can be used to find similarity between words and the visual concepts. For example, the concept ”car” is detected and the description of the concept states that it is an motorized vehicle. These words can then be linked to the concept ”car”. To improve the linking of words to concepts all similar words are added from a lexical database such as Wordnet [12].

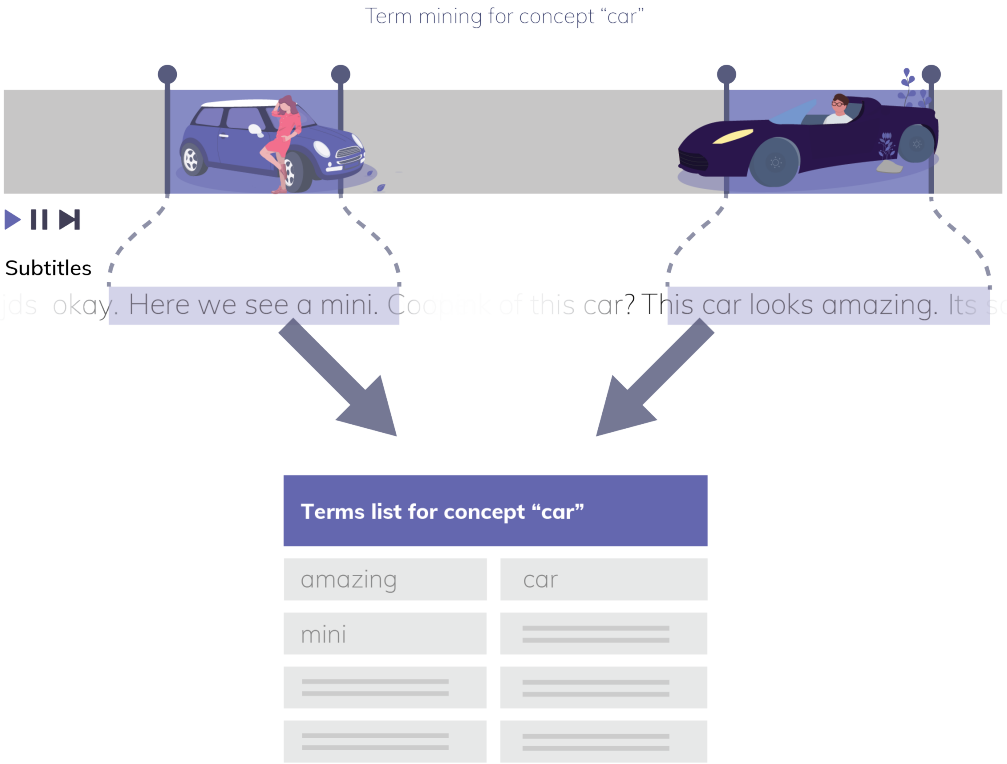


Figure 2.3: Mining associated terms in subtitles through co-occurrence.

A second method uses a statistical approach which mines words in the neighbourhood of the occurring visual concepts as illustrated in figure 2.3. For example, a car is detected as a visual concept, then the words in the subtitles within a fixed temporal neighbourhood around this visual concept are assumed as related words to the concept. These related words are then in turn used to expand the query. Natsev et al. report that the statistical method outperforms the lexical approach.

Sun et al. [13] present a query dependent highlight detection method that also makes a translation step from the visual to the semantic domain. This method uses “viralets”, a mid-level representation bridging between semantic and visual spaces. They created their viralets using a viral video database that consisted of videos and user comments. They grouped visual similar concepts together into viralets and learned the associated semantic terms from the comments of those videos. Those learned terms are then used to match queries to viralets and then viralets to videos. This approach is very similar to the process of Natsev et al. [11] that tries to find semantic terms to expand the query.

The work of Sun et al. did not consider the temporal window in the video sequence selection process. Their method relied on video highlight detection to select the video sequences. From this set of video sequences, the sequence that best matched to the query was returned. The problem is that existing highlight detection methods often suffer from expensive supervision requirements, where human viewers must manually identify highlights in training videos [14]. These highlights are often related to the content of the video and preference of the audience [15].

2.3.1 Word embedding

To improve the semantic term matching between queries and visual concepts, word embeddings could be utilized. Word embeddings are vector representations of words that represent a words context and are used to efficiently measure semantic word similarity. This works different than a lexical database such as Wordnet [12]. A lexical database is manually curated and documents relations between grouped synonyms (synsets) of words. Word embeddings are trained on a large text corpus. The word embedding model looks at all the appearances of a word in this corpus and learns the context from the neighbouring words and stores this knowledge in a word embedding vector. A very simplified example would be that the model comes across the usage of ”king” and ”queen” in the same sentence very often. Then these words must be quite similar. The model learns relations between words in a vast number of dimensions and encodes this in a vector, the word embedding.

A visual example of word embeddings is illustrated in figure 2.4. These are a number of 50 dimensional vectors visualised using red as a positive value (1.0) and blue as a negative value (-1). In the illustration can be seen that the vectors of words like ”woman”/”girl” and ”boy” are somewhat similar. Where the ”water” vector is less similar to all the other vectors in this example. It is important to note that the values in these vectors do not represent a real world feature like probabilities for every word in the vocabulary. These vectors are relations learned by the model expressed in a

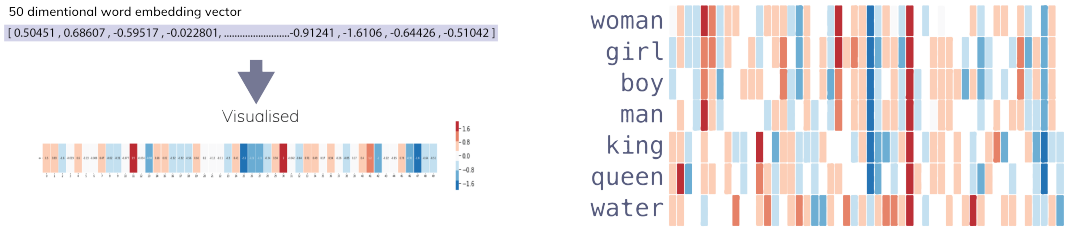


Figure 2.4: Visual example of simple word embedding vectors of a number of words

multi dimensional space.

Mikolov et al. presented Word2Vec [16] [17], a model that can convert single words to the semantic vector space. Other systems such as the universal sentence encoder presented by Cer et al. [18] work on a sentence level instead of single words. These different word embeddings can be used to match the text in the articles with either text learned from ASR or the text describing the visual concepts.

2.4 Segmentation

To be able to perform video clip selection a video needs to be segmented into clips. These clips can then be ranked using associated temporal data. There are different segmentation methods that could be utilized. Videos can be segmented into smaller parts called frames, shots and scenes. Video frames are the most basic unit of video segmentation. A frame is one still image that makes up the series of images that compose a video. A shot can be defined as a contiguous, unedited sequence of frames with start and end boundaries. Scenes consist of one or more shots that are similar in terms of space, time or content.

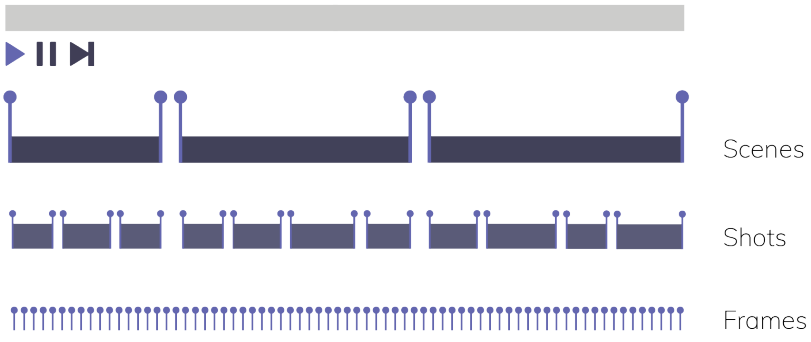


Figure 2.5: Video segmentation into Scenes, Shots and Frames

2.4.1 Shot segmentation

A common way of shot segmentation is using shot boundary detection. This makes use of the changes in frames between shots. There are several ways to change from one shot to another. The most basic way is a direct or hard change, directly going from one shot to another, often referred to as direct shot boundary detection. These abrupt changes are relatively easy to detect by looking at frame similarity. When the next frame is noticeably different from the current frame a shot change is detected. The problem lies with transitions, these come in numerous forms and change one shot gradually to the next shot. For example, a dissolve to black or a wipe effect across the screen to the next shot. Since shot change happens gradually, frame similarity might not detect these changes.

Ngo et al. [19] proposed a system that detects shot boundaries by looking at temporal slices across multiple directions. This means that a sequence of frames is analysed by taking certain pixels from the frames and see how they change over time. These pixels are taken in a cross-hair like form from a video frame. Features can be extracted from these temporal slices to feed into cut, wipe and dissolve detectors.

Other methods [20] [21] use both global and local visual descriptors to measure frame dissimilarity. Using this dissimilarity measure shot boundaries can be detected. For global visual descriptors, RGB or HSV histograms are used. For the local visual descriptors, both methods use SURF [22]. SURF is an interest point detection-description scheme which robustly and reliably finds 'interest points' in the image, such as corners, blobs, and T-junctions. Global features represent the visual content on a higher level. For example, global features can be used to identify scenes through clustering due to the visual similarity among video frames in the same scene. Local features represent the local details of visual content. This is great for finding shot boundaries since these changes are often more subtle that global features might not pick up. Both global and local features can be used to measure the similarity between two frames. These similarity measures are then used to detect shot boundaries by looking at rapid change.

2.4.2 Scene detection

Segmenting in shots negates the contextual sequence information layer. Grouping shots together that show the same content, space or time will form scenes. Scenes include the sequence information layer but are harder to detect. Shots with similar content, space or time need to be clustered together.

There are two different approaches for scene detection, this has to do with the definition you choose to use for a scene. You can say that a scene is a group of

shots that show the same content, space or time, spread over the complete video, ignoring sequence. For example, a series of shots of an apple on a tree are shown, followed by some different shots of eggs and then some more shots of that apple. Using this definition all these apple shots are taken together ignoring the fact that they do not sequentially follow each other. The other scene definition does take the temporal dimension into account and states that scenes are sequential shots with similar content, space or time. Ignoring the temporal dimension seems easier because this makes scene detection a clustering problem, including the temporal dimension makes it a more complex optimisation problem. You could include the temporal position of the clip as one of the features in the clustering problem, but weighting that is not necessarily easy.

Odobez et al. [23] assume that scenes do not have a temporal dimension and treat scene detection as a clustering problem. This method groups shots with similar content into a scene using spectral clustering which allows them to automatically select the number of clusters (scenes) that present themselves in a video. They chose to ignore the temporal dimension because they work in the context of home-made videos. Other methods [24] [25] [26] that do include the temporal dimension use grouping algorithms which group sequential shots optimally to form scenes. All these methods report some issues and there is no standardized test set for scene detection, so the methods are not compared to each other.

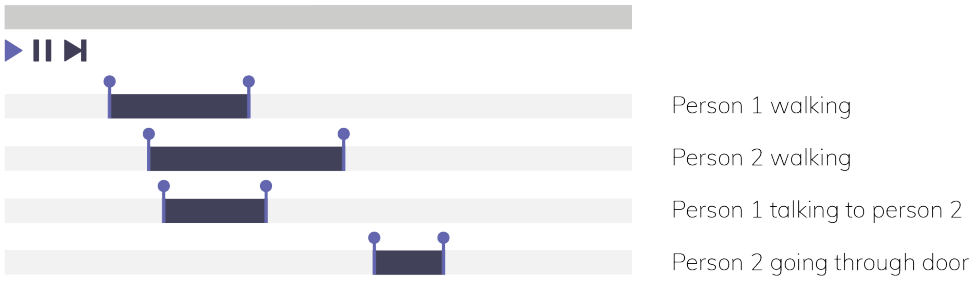


Figure 2.6: Example of stratification of a video

2.4.3 Stratification

Another way to segment videos is stratification [27] as illustrated in figure 2.6. Instead of segmenting the video into shots, the video is segmented into strata. Strata are a contiguous set of frames. Each frame can have multiple strata assigned to them. Meaning that these strata can overlap each other. A stratum represents a

contextual event, for example, a shot begins and ends, the camera zooms in, a character enters and sits down. Using shots as the level of segmenting might compromise contextual information of a video sequence [4]. A set of two shots or scenes might be more relevant together than those shots or scenes separately. By overlapping the segmentation using strata, the risk of compromising the contextual information is lower [4]. Unfortunately, the only presented techniques for this stratification method need to be performed manually and is not yet automated.

Chapter 3

Method: Video Clip Selection System (VCSS) design

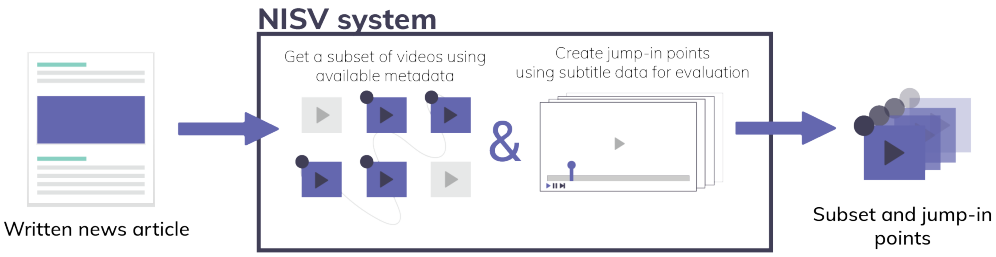
To be able to test the research question, a clip selection system is needed that includes visual features in the clip selection process. This chapter describes a VCSS prototype design that extracts objects from videos to include visual features. The objects, speech data and query are transformed to the same semantic feature space which allows for an easy comparison of video clips to the query. Through this comparison a ranking is build of segments that are closest to the query in this feature space.

3.1 VCSS prototype system overview

Different parts of the VCSS are covered in this chapter. To give a more comprehensible overview this section will describe the process from giving the VCSS system a written news article to giving back selected video clips for that article. Each step in the process is described and visualised with an illustration. Each step has a section associated with it that will go in detail.

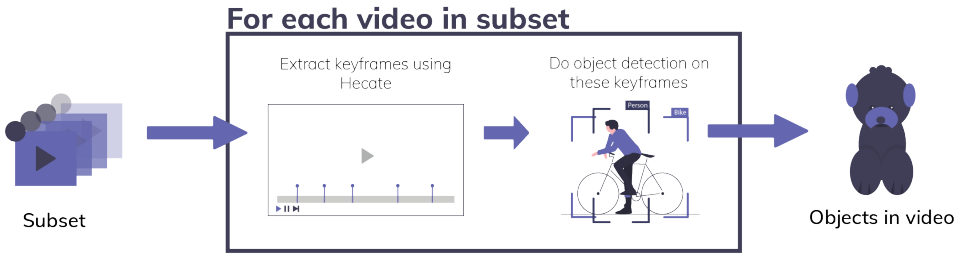
1. **Baseline system**

   The first step is sending the news article to the NISV baseline system. This does two things, it will create a re-ranking subset of videos and it will send back jump in points that can be used as baseline video clips. (Section 1.3)



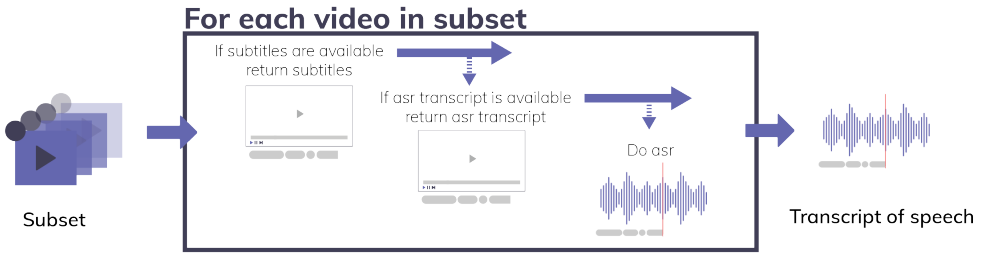
2. **Object detection**

   Object detection is performed on each video to add visual features. This process is sped up by only using extracted key frames from each shot in the video. (Section 3.2)



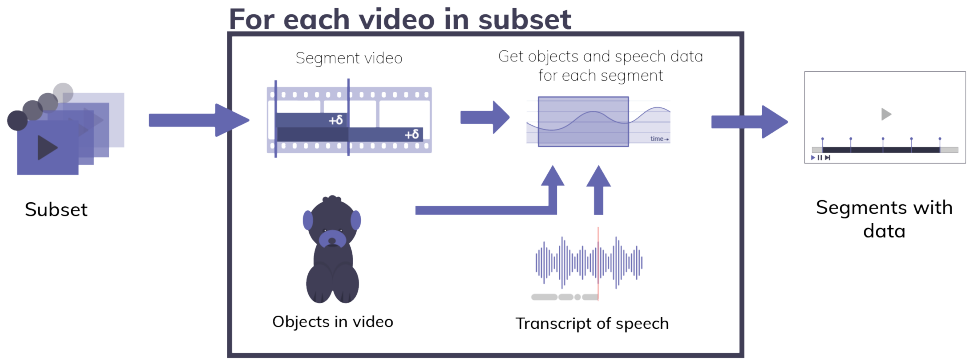
3. **Speech data**

   Parallel to the object detection the speech transcripts are gathered. If subtitles are available this will be used as the speech transcript. If subtitles are not available a transcript is used that is generated by sending the video through an ASR system. (Section 3.3)



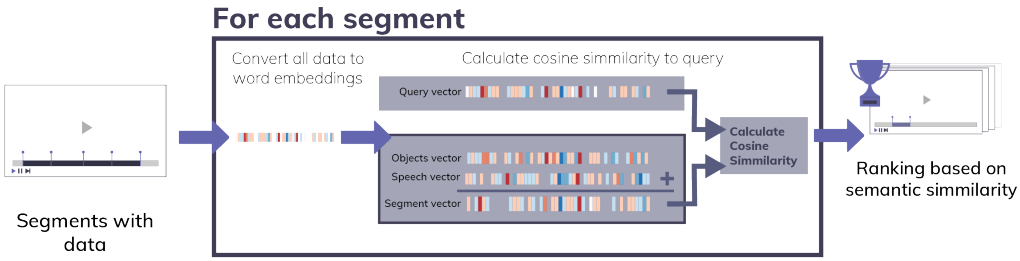
4. **Segmentation**

   After all the features are gathered the video is segmented using overlapping moving windows and a variable window size. This creates segments that overlap each other and have different lengths. For each segment the corresponding speech and object data is gathered. (Section 3.4)



5. **Modality fusion and clip selection**

   The temporal data is converted into word embeddings. The object data and speech data are then combined into a single vector. The news article that is used as input to the system is converted to word embeddings as well. The fused speech and object vector is compared to the news article vector using cosine similarity. This results in a similarity score for the segments. Finally all the segments are reordered using the similarity score creating a ranked list of segments. The top of this list can be used as the output of the system. (Section 3.5)



3.2 Extracting visual features

Concept detectors were chosen as the starting point for extracting visual features as they are frequently used and provide contextual information.

3.2.1 Concept detection

Concept detectors deal with detecting instances of semantic objects (such as humans, buildings, or cars) in images and videos. Videos are processed by treating individual frames as images for the concept detectors. Concept detection is generally done using machine learning. These machine learning models learn patterns connecting concepts to features extracted from an image. To be able to train these machine learning models a dataset of images is needed where each image is annotated with the concepts it contains. In the training process each image is fed to the machine learning model to see if it detects the right concepts. If the detected concept does not match the annotated concept, then the parameters of the model will be adjusted. This is repeated until the model performs satisfactory. A concept detector has a predetermined number of concepts it can detect. This is determined by the number of concepts the training data contains. For example, when the model is trained with images of only cats and dogs, the final model will only be able to detect cats and dogs.

3.2.2 Chosen Model

The chosen criteria for a concept detector for the clip selection system are:

- A detector that detects a broad number of concepts (paragraph 3.2.3)
- Object detection (paragraph 3.2.4)
- Pre-trained model (paragraph 3.2.5)

There are many pre-trained models out there. For this system a model from the Tensorflow model zoo[1] was chosen. The models in this collection were tested for speed and accuracy trade-offs by J. Huang et al. [28]. The VCSS prototype system will not work in real time, so speed is negated as a choice factor. The models trained on the Open Images [29] dataset were chosen as the best options, since they could detect the most and diverse concepts with comparable accuracy to the other models. The exact pre-trained model used is *faster_rcnn_inception_resnet_v2_atrous_oidv4*

[1] Tensorflow Model Zoo `https://github.com/tensorflow/models/blob/master/research/object_detection/g3doc/detection_model_zoo.md`

which is the model with the highest accuracy (measured in mAP[2]) that is trained on the Open Images dataset. Open Images contains nine million images of which 1.74 million are annotated with bounding boxes. These bounding boxes makes the images usable for training an object detection model. The object detectors trained on this dataset are able to detect 600 concepts which are part of a semantic hierarchy which is illustrated in figure 3.1. These semantic relations can be utilised as touched upon in section 2.2.

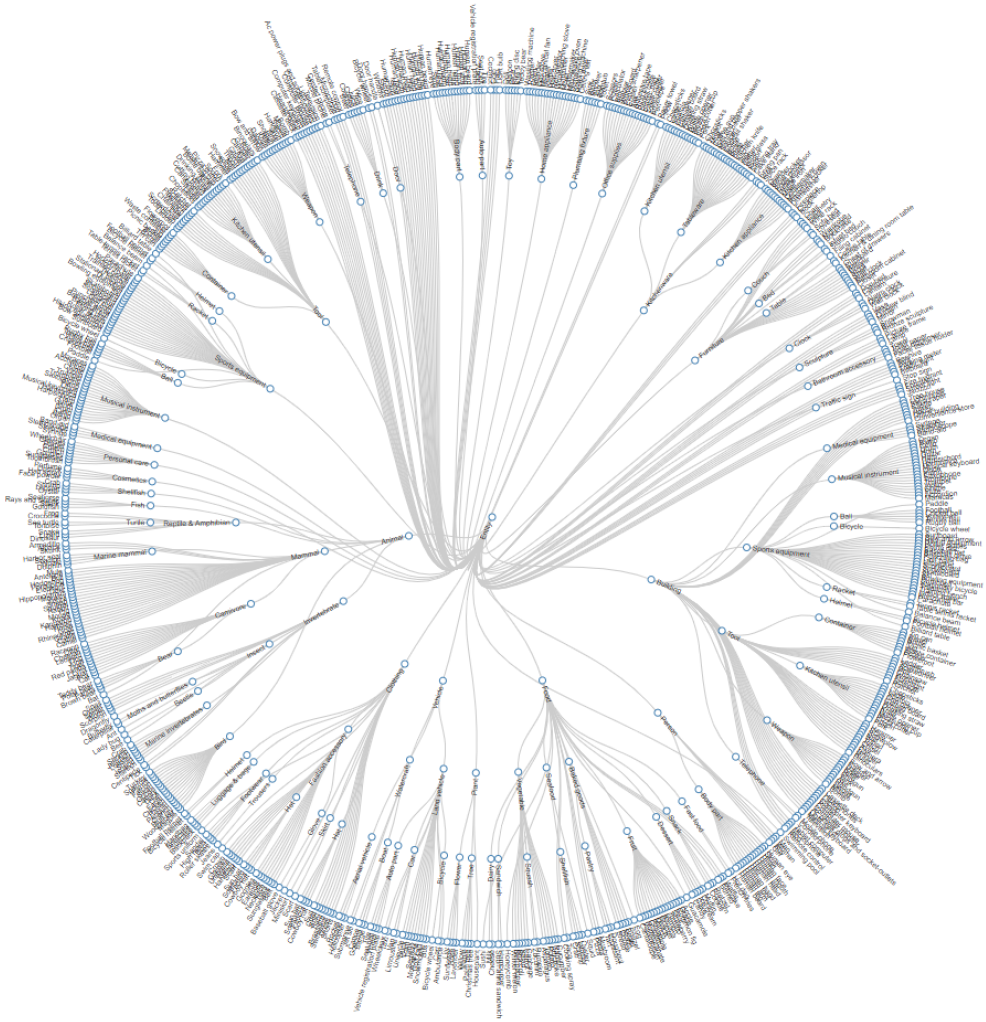


Figure 3.1: Open Images object classes with their semantic hierarchy.
can also be found online: Open Images semantic hierarchy - `https://storage.googleapis.com/openimages/2018_04/bbox_labels_600_hierarchy_visualizer/circle.html`

[2]Open Images mAP = `https://storage.googleapis.com/openimages/web/evaluation.html#object_detection_eval`

3.2.3 Specialised vs Broad

A concept detector can be highly specialised or very broad. Examples of highly specialised concept detectors are face detection and ball tracking during football matches. More broad detectors are generally called object recognition or object detection. These detectors often have a wide range of concepts they can detect. The advantage of highly specialised systems is that they are often more accurate. They do not have to generalise the model to recognise a lot of concepts, instead these models are more fitted to a small list of concepts. It is assumed that is favourable to have many possibilities to match a concept to a query. For example, when a concept detector only can detect faces it would not provide much contextual information other than that there are persons in the image. With a wider range of detectable concepts, a broader range of conceptual information can be extracted. Unfortunately increasing the number of detectable concepts impacts the performance of the detector. More concepts often has an impact on the performance and accuracy. You need a more complex model and more training data. With a lot of concepts there can be some concepts that are very similar and hard to distinguish for the detector. A choice is made for a broad as possible detector that has acceptable performance for each concept.

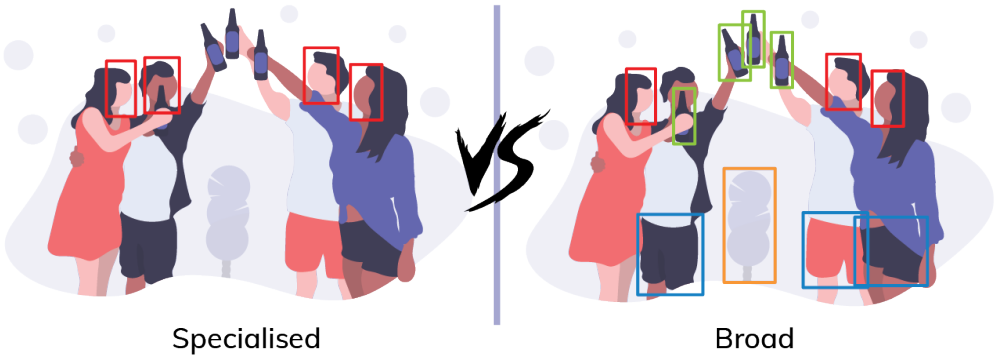


Figure 3.2: Specialised and broad concept detectors.

3.2.4 Object detection vs Object classification

The general term for broader detectors is object detection, but there is a very important difference between object detection and object classification. They both can tell you what object is in the image. The most important difference is that an object detector can tell you where an object is in the image. An object classifier can not do this. A visual comparison can be seen in figure 3.3. An object detector always

contains an object classification step. An object detection model first ”looks” where there might be possible objects, then these possible objects are fed to an object classification model and this tells what the object might be. These two models together provide a location and classification of the objects in an image. The object locating step is important to stop feeding conflicting information to the object classification model. For example, when an image of a cat and a dog is fed to an object classification model directly it gets confused and gives a high probability to the concepts of both cat and dog. With only two concepts this is still manageable, but with more concepts in the image it may confuse the model and results in lower probabilities of these concepts. Another difference is that object classification is not able to detect multiple instances of the same concept where object detection is able to do this. For example, when given an image of two cats, object classification would classify the image as a single cat.

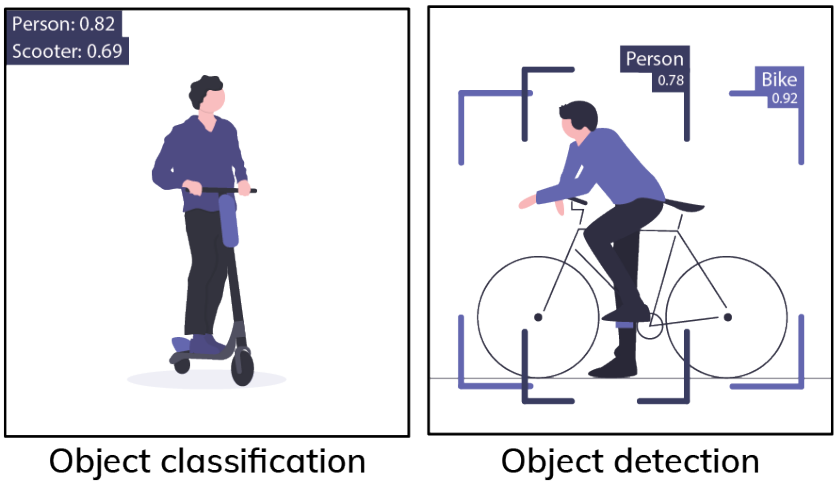


Figure 3.3: Object classification shows what objects are in the image, object detection also shows where in the image the objects are.

Adding a region proposal algorithm that ”looks” for possible objects, each object is fed to the classification model separately, so the information is less conflicting. This improves the performance of the classification step for images with multiple concepts, but adding this region proposal algorithm increases complexity of the system and introduces an error in to the process. This introduced error results often results in a lower amount of concepts able to detect. Object classification can detect a higher number of concepts reliably. An assumption was made that object detection is more suitable for the clip selection process, since the videos almost always contain multiple concepts and multiple instances of these concepts. Knowing that there are multiple instances of a concept in an image could help gauge the importance of

that concept. For example, if there are multiple lions in an image the context would probably be lions, where a single lion might suggest a broader context like savanna animals.

3.2.5 Pre-trained models

When there is no time and resources available to train a model, pre-trained models can be used. These pre-trained models are created and trained by someone else on a dataset often openly available. Images are a universal medium, so a pre-trained model can be easily used on the NISV data. Pre-trained models that perform well are easily found, but these models are not validated on the dataset of the clip selection system. These models are often validated on openly available test data. There might be some deviations in the clip selection dataset that will not be handled well by the pre-trained models. Using the pre-trained models is the only option for this system, so these possible deviation need to be taken in consideration during evaluation. The clip selection system uses pre-trained models because training a model ourselves is not in the scope of this research.

3.2.6 Adding visual features on the fly

The NISV video archive contains a lot of data, too much data to process in a short time. To add visual features to this archive and still perform retrieval in a reasonable time, a multi-media re-ranking approach [5], [30] has been used. This process is illustrated in figure 3.4. This re-ranking approach selects a subset of videos for each query using the currently available features and metadata. Then visual features are added to this subset on the fly and finally, this subset is put through the segmenting and ranking processes.

If all the visual features would be present in the dataset beforehand, the method of segmenting every video and ranking each segment individually will not be viable. The process of comparing the temporal data of every segment of every video will take a long time. Other methods of comparing the temporal data to the query need to be found. Re-ranking might miss some relevant documents in the first pass, but is much faster than adding visual features to all the videos in a database.

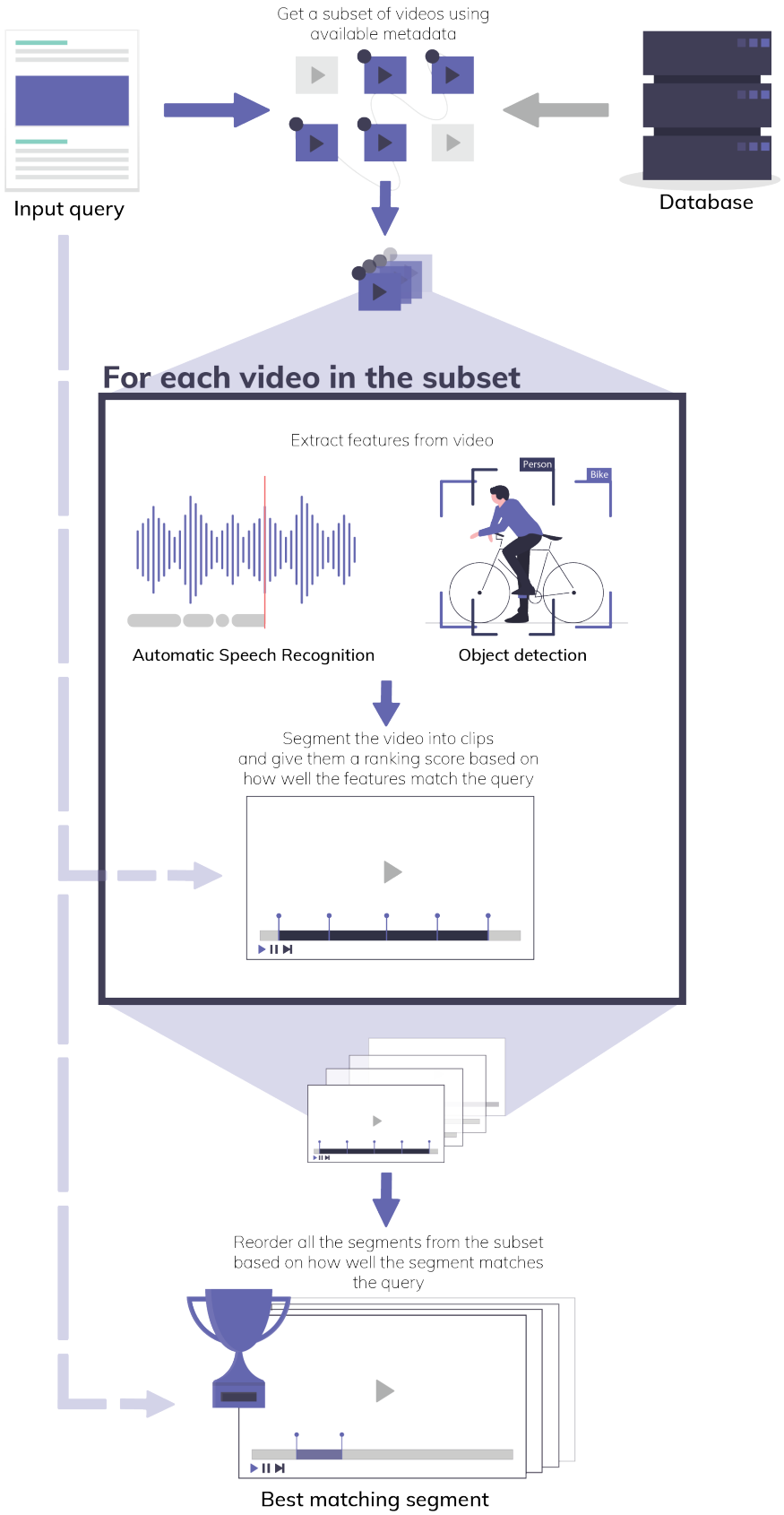


Figure 3.4: Re-ranking solution for the VCSS

3.2.7 Speeding up object detection

Doing object detection on every frame of a video is very time consuming. For an average video this means that per second of video 24 images need to run through the object detector. A simple approach to speeding up the detection process is only using one frame every second. This speeds up the process 24 times. Another method is using shot segmentation, then only one frame per shot needs to be detected since all frames within the shots are very similar. To select suitable key frames a method presented by Song et al. [31] was used. After segmenting, certain key frames are selected based on the amount of motion in the frame. Key frames with less motion in them are desirable since they contain less motion blur and produce a clearer image for the object detector to work with. The motion in a frame can be measured by looking at motion features. Segmenting the videos adds time to the overall process, but this time is regained when only one frame per shot is used for the object detector. With the added benefit that these frames contain the least amount of motion.

3.3 Extracting speech features

Some videos in the database had subtitles, some had ASR transcripts and some documents had no speech data at all. The documents for which neither a subtitle or an ASR transcript was available an Dutch ASR system was used called KaldiNL[3] to generate the missing ASR transcripts. This is the same ASR system that was used to create the existing ASR transcripts in the database. If available, the subtitles were used first for the clip selection process. Otherwise the ASR transcips were used. Manually annotated subtitles are favourable over automatically generated speech transcripts since they are often more accurate.

[3]KaldiNL ASR system - `https://github.com/opensource-spraakherkenning-nl/Kaldi_NL`

3.4 Segmentation

A logical first step of finding the clip that matches the query is segmenting the video. These segments can then be compared to the query individually returning the best matching segment as the best matching clip. A number of segmentation algorithms were explored. Scene segmentation is still in early stages of research and the openly available algorithms produce no coherent scenes, so it was not considered as a viable option. Multiple shot segmentation methods were tested and produced viable segmentation. During the testing of these segmentation methods a realisation was made. Video shots have different lengths and different information densities. The temporal data is not uniformly dense over the video. For example, certain parts of the video might not contain any speech data and other parts might not have any detected objects. A combination of sequential shots can be a better match than using only one video shot. The shot boundaries are logical start and stop points in the video, but also divide up the temporal information. Since the focus of this research lies on the addition of the visual modality in the clip selection process, a decision was made to keep the segmentation process simple and uniform.

The chosen segmentation approach takes inspiration from stratification (see section 2.4.3). Using stratification the segments do not have to be successive windows, they can also overlap each other. A choice was made for a segmentation algorithm that uses multiple overlapping moving windows of varying window lengths. This segmentation method considers clips that can transcend shot boundaries and different lengths of the same content to accommodate different information densities.

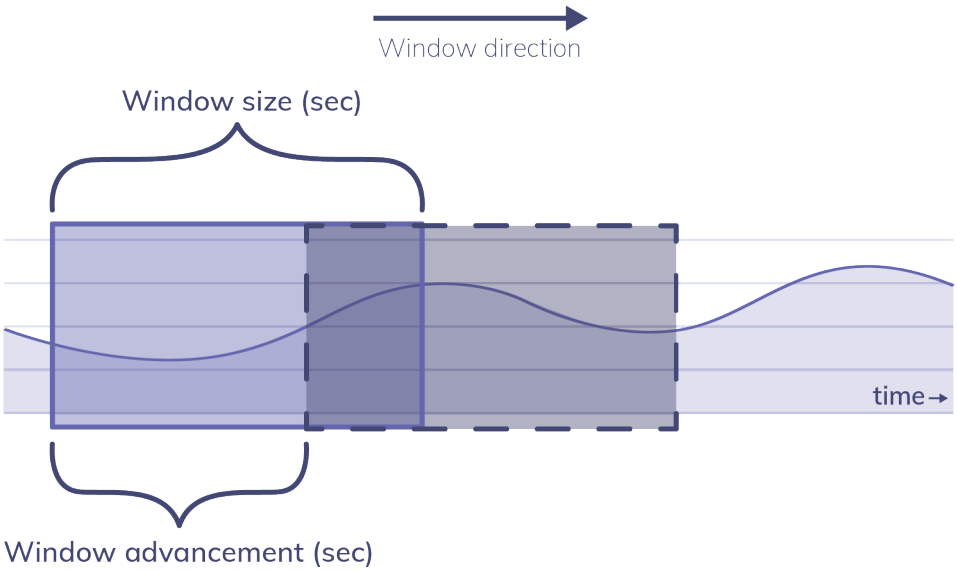


Figure 3.5: A moving window over temporal data

For a moving window there are two variables to consider, the size of the window and the amount that a window advances each step. These variables are illustrated in figure 3.5.

Overlapping the windows generates more possible clips. If there is no overlap, a situation could occur that the best possible clip starts at the end of one window and stops in the next window.

Window size has influence on how well the clips match to the query. A longer segment might provide a better match to the query than when it is split up in smaller segments. To include the short as well as long clips, the videos are segmented multiple times with varying window sizes. A minimum and maximum window size is determined. The process starts by segmenting the whole video using the minimum window size. Then the window size is increased with a fixed amount of seconds and the video is segmented again using the new window size. This process is repeated until the maximum window size is reached.

This segmenting process has four variables, the minimum window size, the maximum window size, the window advancement and the window size increase. To simplify the segmenting approach the window advancement and the window size increase are combined in a single δ variable, because they both control the spread of the segments over the video. The segmentation process is illustrated in figure 3.6.

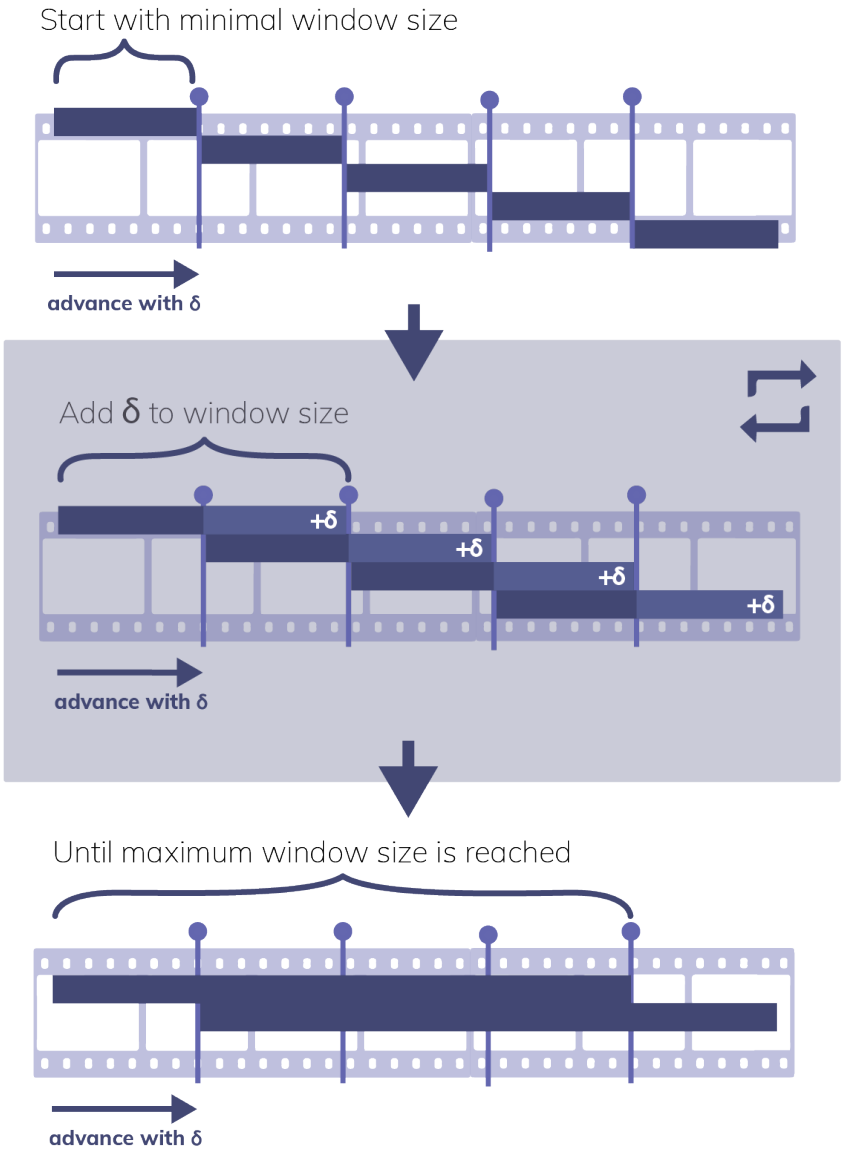


Figure 3.6: Segmentation process using an overlapping moving window and variable window size.

3.4.1 Temporal data and Temporal windows

To be able to compare a segment to the query the speech and object data associated with that segment needs to be retrieved. This is easily done since both the object and speech data are temporal. Temporal data in this context is defined as data of which is known at what timestamp in the video it was recorded. For example, subtitle data tells what is said in the video, but more importantly, when it is said. This temporal data is already present in the dataset used for this research in the form of subtitles or a speech transcript. Additional visual temporal data is

added using object detection. The object detection process annotates where and when these objects occur in the video. To easily get corresponding temporal data of a video segment a temporal window can be used. This creates a subset of the temporal data containing only the datapoints that occurred in the specified window of the video segment as illustrated in figure 3.7.

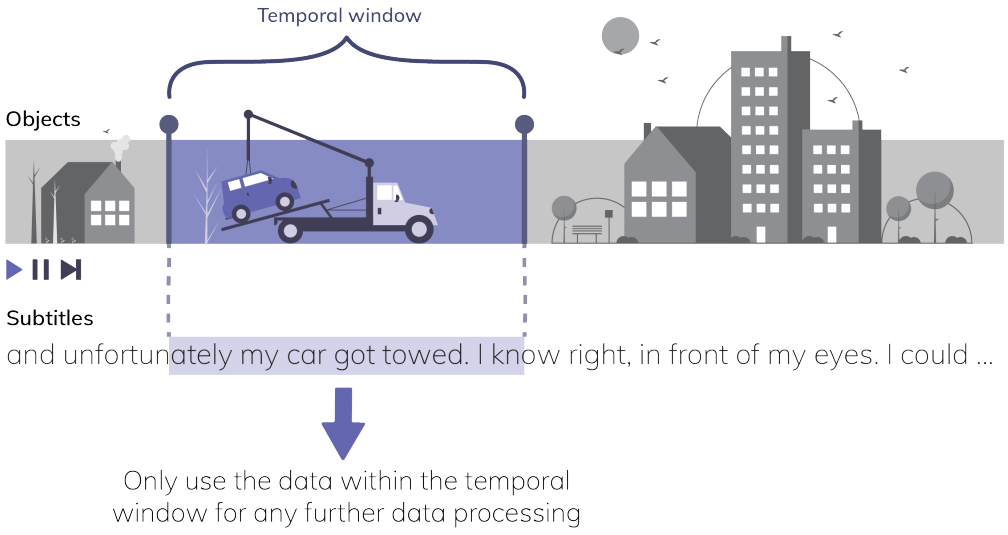


Figure 3.7: A temporal window creates a subset of datapoints from a specific time in the video

3.5 Selecting a clip

3.5.1 Parameters

There are a number of parameters that can be tweaked to alter the clip selection process. During segmentation the minimum and maximum lengths of the segments can be altered. As well as the δ variable that controls the overlap and spread of the segments. The influence of the modalities on the query comparison can be altered by increasing or decreasing their corresponding scalars. This leaves five parameters to be tweaked to find an optimum for the VCSS.

- Minimum clip length
- Maximum clip length
- δ overlap
- Speech vector scalar
- Object vector scalar

The final parameters values were chosen through experimentation. Unfortunately the resources to test multiple versions of the system were not available. So no hyper parameter tuning was not possible. The experimentation to find the best parameters consisted of running tests on a defined set of news articles, changing parameters and reviewing the video clips.

The minimum, maximum and δ variable were chosen through domain knowledge and common sense. These were set to five, thirty and five seconds respectively. This is a reasonable range of clip lengths and overlap that would not extremely over or under sample the videos.

Selecting the scalars of the modalities was a more ambiguous process. Multiple test runs were done with the scalars set at extremes to find a range where the system performance was acceptable. Unacceptable performance would be that most clips selected by the system have nothing to do with the news article or the clips are mostly uninteresting to look at. An acceptable range was found around a 2 to 1 ratio between the modalities both ways. A final experiment was done were the news articles of the final experiment were run through the system with three different settings for the scalars. First with 2 for the speech scalar and 1 for the object scalar, second with the scalars being equal and last with 1 for the speech scalar and 2 for the object scalar. The top three clips for each news article from each run were put next to each other. For each article one scalar setting was chosen as the best and got a point. This scoring was done by two separate people. The parameter setting with the highest number of points was selected as the final parameter setting. This ended up being 1 for the speech scalar and 2 for the object scalar.

The final parameters can be seen in table 3.1.

Parameter	Value
Minimum clip lenght	5 sec
Maximum clip lenght	30 sec
δ overlap	5 sec
Speech vector scalar	1
Object vector scalar	2

Table 3.1: Final clip selection parameters

3.5.2 Translating features from the visual domain to the semantic domain

The visual features extracted by the object detector can not be directly compared to the words in the query. Mid-level representations can be used to translate visual concepts from the visual to the semantic domain [11] [13]. This is done through learned semantic term lists associated with every concept. These term lists are generated using co-occurrence. If a concept is detected, the semantic terms that occur within a temporal window are gathered and added to the list. This can be from comments, metadata or subtitles. An assumption is made that the most co-occurring terms have a strong link to the concepts.

3.5.2.1 Co-occurrence term mining

The subtitles and speech transcripts of the videos were used to extract semantic terms that have a strong relation to the concepts. Subtitles and ASR transcripts have temporal information that allows to find semantic terms within a close time window of the occurrence of the concept. For example, when the concept ”car” is detected all the words that occur five seconds before and after this concept are added to the term-list of the concept car. This process is illustrated in figure 2.3. When words are added to the term-list every word is made lowercase and the stopwords are removed. Stopwords are frequently occurring meaningless words like ”and”, ”the”, ”is” etc. The stopword list that was used for this system was acquired through the Natural Language Toolkit[4] for Python. After processing the videos and their transcripts each concept has a list of terms that co-occur frequently.

[4]NLTK - `https://www.nltk.org/`

3.5.2.2 Experimentation with term mining

A number of co-occurrence term mining experiments were performed. The detection window around the object occurrence was varied in between experiments to see how much it influenced the term lists of the concepts. An observation was made that common concepts, like ”person” or ”face”, were matched with almost every word from the speech transcript. The link between terms and concepts could not be established correctly because of this. Early tests were done on a small dataset since the object detection of the videos did take a significant amount of time. Mining more data would not solve this problem since the occurrence frequency and context of these concepts does not change. For other more less general concepts the co-occurrence mining seemed to work. For example, the concept ”syringe” was matched with terms like ”hospital” and ”nurse”. By co-occurrence mining we hoped to overcome the problem of double word meanings. For example, the word ”book” can mean the physical book you can read or it can refer to booking a ticket. When using co-occurrence mining words like ”reading” and ”story” might co-occur with the concept ”book”, steering the word meaning in the right direction. If this actually is the case needs to be researched further. Because of time and resource restraints the choice was made to take a more simple and robust manual term matching approach.

The pre-trained object detector that is used in the VCSS is trained on a dataset that has English labels. A translation to Dutch was needed to match these labels to the Word2Vec representations. An automatic translation step was introduced to speed up the mapping process. This translation step was done using WikiData[5], which is a knowledgebase of concepts. Each concept of the object detector has an id that can be linked to a WikiData item. These WikiData items often contain a description of that item in Dutch. An automatic process would suggest a Dutch word from the description to the user which could accept it as the semantic term for the object or overrule it by adding a word manually. This was done for all 600 concepts of the object detector.

3.5.3 Bringing all features to the same semantic feature space

The system needs to be able to rank video clips on how well they match with the query. Traditionally this is done using the Term Frequency (TF) and Inverse Document Frequency (IDF) [32] on metadata. Both these metrics look at word occurrence in documents. In this case the documents would be the ASR/subtitle data and the object terms. TF and IDF are both metrics for word occurrence, so the searched words need to appear in the documents. For example, when querying for the word

[5]WikiData - `https://www.wikidata.org/wiki/Wikidata:Main_Page`

”music” a document needs to hold the word ”music” to get a positive ranking score.

Each visual concept is manually mapped to one term, this leaves a ”vocabulary” of 600 words (because there are 600 concepts). This relative small vocabulary can be a problem for the TF-IDF approach since the chance of the words in query matching with the terms from the concepts is small. Another potential problem is that we want to match clips to the query. The ASR transcript of a 30 second clip does not hold a lot of text. A small body of text has a small chance of matching the words in the query exactly. The ideal scenario would be that documents that have words that are very close semantically to the query get a high rating as well. A way of doing this is matching on the semantic level using word embeddings. For example, a search is done using the word ”trousers”. The word ”trousers” needs to occur at least once in these documents to be a good match to the query for a TF-IDF systems. When using word embeddings all the documents that have words which have a close semantic meaning to the query, like ”jeans” and ”pants” would be considered a close match as well.

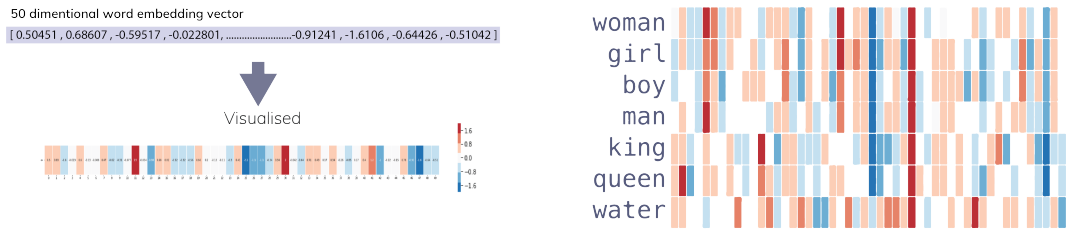


Figure 3.8: Visual example of simple word embedding vectors of a number of words

Word embeddings are vector representations of words that represents a words context. These embeddings can be used to efficiently measure semantic word similarity. A word embedding can be created using multiple methods, but the Word2vec [16] method is chosen because of successful implementation in different fields and the wide availability of pre-trained models. Given enough data, usage and contexts, Word2vec can make highly accurate guesses about a word’s meaning based on past appearances. Those guesses can be used to establish a word’s association with other words (e.g. “man” is to “boy” what “woman” is to “girl”). A visual example can be seen in figure 3.8.

A pre-trained Word2vec model was acquired from Coosto[6]. The model holds only Dutch words, since the used video material for this research is all in Dutch. The acquired model was trained using 600 million individual messages, comprised of Dutch social media messages (624 million messages) and Dutch news, blog and fora posts (36 million messages). The exact training method is described on the

[6]Dutch Word2vec model - `https://github.com/coosto/dutch-word-embeddings`

Github page of Coosto. The pre-trained Word2vec models can be seen as a dictionary, you give it a word and the model returns a vector representation of that word. With these pre-trained models there is a possibility that a word is not in the model. For this implementation of the VCSS these unknown words are ignored.

The object detection model holds a relative low amount of concepts, so each concept is manually matched to a single word embedding. For example, the concept ”car” is matched to the Word2Vec representation of ”car”. The word embeddings for the words from the ASR transcript and subtitles can be automatically retrieved from the Word2Vec model, so no manual mapping is needed there.

Word2Vec works on a word level, so when sentences need to be semantically compared to each other each word in each sentence is individually converted to their vector representation. Then these vectors can be added to each other to create a vector that summarises the semantic meaning of the complete sentence as illustrated in figure 3.9. The vector does not have to be averaged over the amount of words in a sentence since magnitude is not considered in the similarity measure. However, it is advised to normalise vectors before adding them to vectors that come from different systems or modalities.

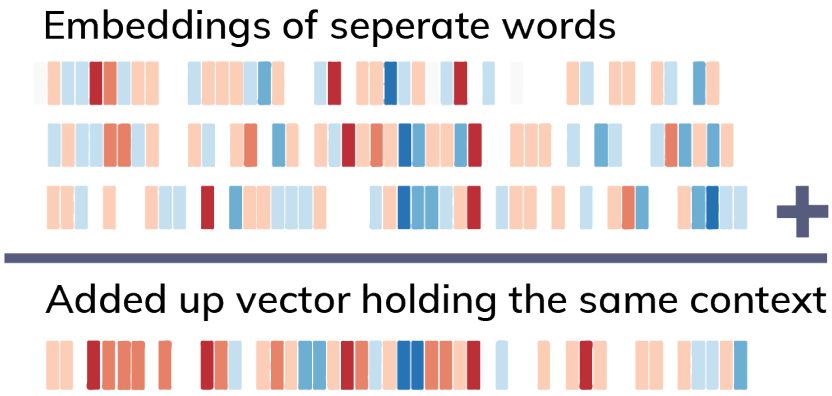


Figure 3.9: Adding embeddings of words to get an embedding on sentence level holding the same context

When aggregating words together one word might not be as important as the other. Much used words like articles and auxiliaries do not have as much impact on the context as domain specific words. To make sure these words do not overshadow the semantic meaning on the sentence level a weighting technique is used. C. de Boom et. al. [33] have found that including the inverse document frequency of each word as a weight scalar to the corresponding word embedding better represents the semantic meaning. The inverse document frequency is a widely used metric in

information retrieval. It is a measure of how unique a word is across all documents. It is calculated by taking the inverse fraction of the documents that contain the word and scaling it logarithmically.

This process is done for the speech transcripts and object terms separately. Each term embedding representing an object is multiplied by the IDF of the corresponding object. And each word embedding from the transcript is multiplied by the IDF of the corresponding word.

3.5.4 Measuring semantic distance

The semantic distance between two semantic vectors can be calculated using the cosine similarity. Given two vectors, A and B, the cosine similarity, $\cos(\theta)$, is represented using a dot product and magnitude as

$$\text{similarity} = \cos(\theta) = \frac{\mathbf{A} \cdot \mathbf{B}}{\|\mathbf{A}\|\|\mathbf{B}\|} = \frac{\sum_{i=1}^{n} A_i B_i}{\sqrt{\sum_{i=1}^{n} A_i^2}\sqrt{\sum_{i=1}^{n} B_i^2}}, \tag{3.1}$$

where A_i and B_i are components of vector A and B respectively.

Cosine similarity looks at the direction of the vectors, but not at the magnitude. So two vectors with the same orientation have a cosine similarity of 1, two vectors oriented at 90°relative to each other have a similarity of 0, and two vectors diametrically opposed have a similarity of -1, independent of their magnitude.

3.5.5 Ranking segments

There are two modalities that need to be combined in the ranking process, the speech and the visual modality. There are two mayor ways of doing this, using early fusion or late fusion approach [34]. With early fusion, the extracted features from the different modalities are combined and then used as an input for the system. Using the late fusion approach, the modalities are combined at a later stage. Each modality builds a separate ranking first and are then combined to form a final ranking.

An early fusion approach is chosen for this system, since both modalities can be easily combined by converting them to the same vector space (Word2Vec). This takes away the complexity of multiple ranking models that would be used in a late fusion approach. The early fusion is performed by converting both the speech and object data to their Word2Vec representations. Both these vectors are then normalised and combined. For each modality a scalar is added to control the influence

of each modality on the ranking process. This results in the following formula.

$$\vec{f} = (s1 * \vec{o}) + (s2 * \vec{s}) \tag{3.2}$$

Where $\vec{f}$ is the fused vector, s1 and s2 the scalars, $\vec{o}$ the object vector and $\vec{s}$ the speech vector.

The fused vector can then be compared to the vector representation of the query using the cosine similarity (formula 3.1). The segments can then be ranked using this similarity. This process is illustrated in figure 3.10.

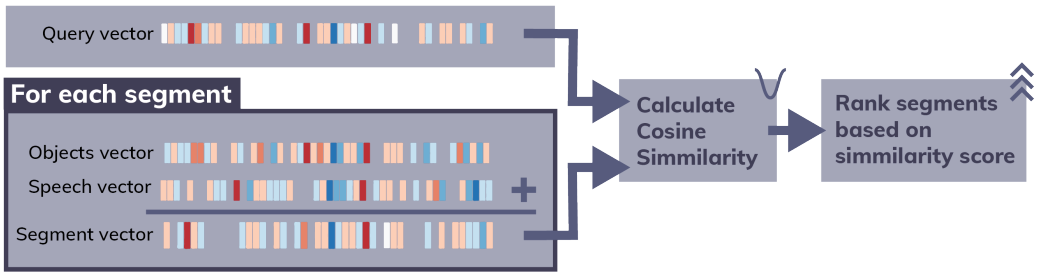


Figure 3.10: Create a ranking score through semantic similarity of query and video segment

An observation was made that shorter clips were often ranked higher. This might be caused by the fact that the longer clips hold more data, and as a cause of that, more data irrelevant to the query. Between relevant parts of a longer clip might be parts that have nothing to do with the query. To give longer clips with relevant parts in them a better chance some experiments were done penalising shorter clips. This was a simple system that lowers the ranking score based on clip length. After a few experiments this addition was reversed since it seemed that it had no impact. This was done through observation, this needs to be tested further on a system with and without this addition to get significant results. The choice was made to keep the VCSS simple since there was no clear performance gain by penalising short clips in the early testing phase.

Chapter 4

Method: Experiment

The previous chapters describe how a video clip selection system that incorporates the visual modality works. It is hypothesised that this system selects clips that have a higher visual relevance and visual interest than a system that does not use the visual modality. This chapter puts this hypothesis to the test.

4.1 Goal and Scope

The goal of the experiment is to verify the two hypothesises. The first hypothesis is that using the visual modality allows us to select more visually interesting video clips. The second hypothesis is that using the visual modality allows us to select video clips that are more visually relevant. In this experiment, we use the system described in chapter 3, which relies on object detection to characterise the visual modality. The proposed VCSS, which uses object detection and speech transcripts for clip selection, is tested against the baseline NISV prototype which uses metadata and speech transcripts to select clips. The clips of both the systems are compared by users to find out which system is preferred.

This experiment focuses on the visual modality, so the audio of the clips is not used for testing. The clip selection systems do not have time restrictions to process the videos. For this experiment the clip selections are pre-processed. The engineering challenge of near real time clip-selection is not part of the scope of this research. This experiment is done with news articles as an input for the selection systems. The videos for this experiment consist of mostly news broadcasts. The systems are compared on the perceived visual interest and perceived visual relevance of the clips.

4.2 Method

Our goal is to evaluate clip selections on:

Perceived visual interest: A visually interesting video is catching the attention of the user by the visuals being unusual, exciting, or having a lot of ideas. For example, a clip of a talking head is less interesting than a clip that explains deforestation with visuals. The second clip provides information and offers ideas, making it more interesting. Catching the attention of the user with visuals is especially important in situations where you cant utilise other means such as audio to inform the user. Improving perceived visual interest will help drawing the attention of users to the video clips.

Perceived visual Relevance: A visually relevant video contains objects, scenes and information connected to the query. For example, a clip that shows a penguin is more relevant to a query about penguins than a clip that shows an iceberg. The first clip actually shows penguins, making it more connected to the query about penguins. Providing information connected to the query is important to provide the user with the expected information they requested through the query. Improving the perceived visual relevance will help providing the user with the demanded information.

Where visual interest is geared towards catching the attention of the user the visual relevance is more focused on giving the user the right information. Both aspects are important to a good functioning retrieval or recommendation system.

4.2.1 Performance evaluation

For the evaluation of query-dependent clip selection are no test datasets readily available. Therefore, the VCSS prototype is compared to an existing prototype system, further called 'NISV prototype'. The NISV system does not use the visual modality in the clip selection process and the VCSS prototype does. The two systems are otherwise functionally similar. This comparison lets us evaluate the performance difference of a system with and without the visual modality.

Assessment experiments are very common to test the performance of retrieval or recommendation systems. The most used assessment method is to grade the documents on a scale. For example, an human assessor is presented with a search query and a set of videos. The assessor is then asked to grade the videos based on relevance to the query on a scale from 1 to 5. This method is useful to assess large datasets, you only need to visit every video once and you directly get a score based

on the grades given.

The definition of the two test aspects can be interpreted differently by multiple people. Giving these video clips a score on a fixed scale might hurt the consistency of the performance measurement. Different fields of study [35]–[38] show that ranking on an absolute scale is a cognitively hard task and different test subjects may interpret vocabulary and intervals of the rating scale differently.

A proposed solution for this problem is a relative ranking approach through pairwise comparison [35], [38]. The videos are evaluated in pairs. An assessor is always presented with two videos and the question which of the two performs better (figure 4.1). If you do this often enough, a pairwise ranking matrix can be built and a full ranking can be derived from that matrix (figure 4.2). With a full ranking is meant that we know for each clip how it compares to all the other clips. If all pairs have been assessed the same number of times, the ranking is derived by counting the number of times a certain video is chosen over the other. In this example video C was preferred two times over other videos. These counts are then ordered from large to small and that forms the ranking of the videos. This shows the major drawback of pairwise comparison, it takes a lot more time. To fill a complete preference matrix you would need $(N(N-1)/2)$ pairwise comparisons. For example, to rank 12 samples you need $(12(12-1)/2) = 66$ comparisons instead of the 12 you needed to rank on a fixed scale.

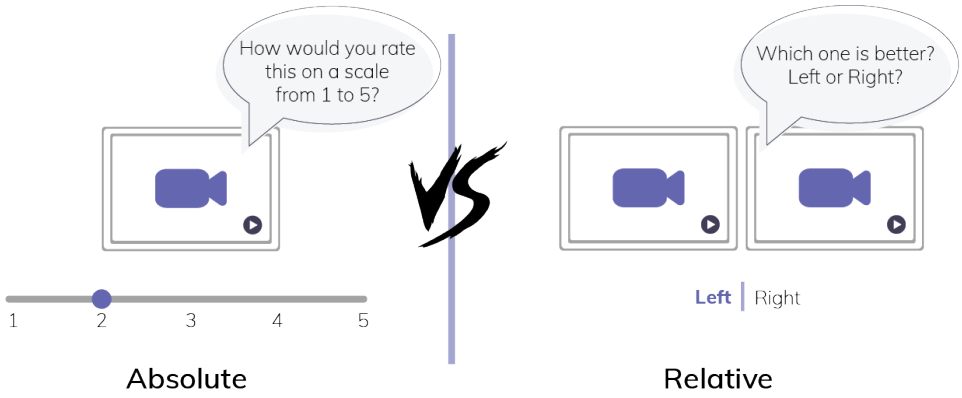


Figure 4.1: Absolute and relative assessment methods

There are methods to reduce the numbers of pairwise comparisons and still build a full ranking. These methods assume transitivity and reciprocity, meaning that the assessors would rank the non-compared pairs the same as the already compared pairs. Linares et al. [39] show that this is not always the case, but simpler, more intuitive problems seem to be less affected by irreciprocity.

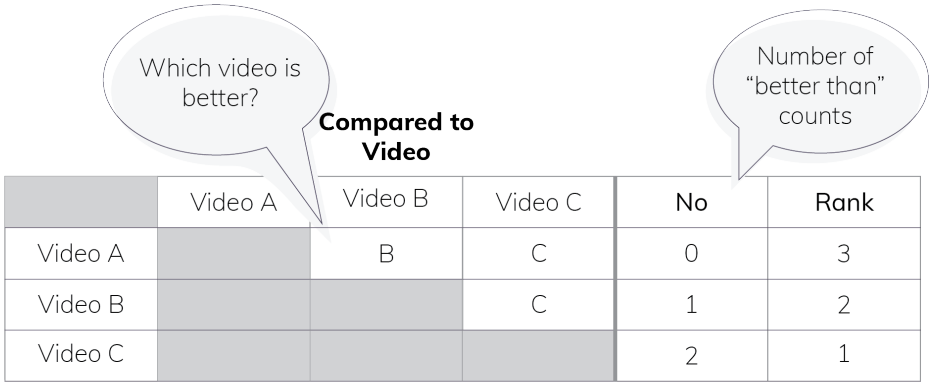


Figure 4.2: Pairwise ranking matrix example

Building a complete pairwise ranking for all the clips selected by the two systems is unfeasible, simply because it would ask to much of the assessors. For example, with only twenty clips from around 30 seconds long an assessor would have to do 190 comparisons. A simpler approach is taken to reduce the amount of comparisons. From each system, a clip from a certain rank is taken and directly compared to the clips from the other systems. For example, the top ranked clip from the VCSS prototype is compared to the top ranked clip from the NISV prototype. To test if these systems perform better than random selection, a randomly selected clip is also added to the comparison. The assessor is then asked to choose one of the three clips that performs best on a certain aspect. In the case of this experiment the aspects would be either visual relevance or how visually interesting the clip is. This proposed method ranks on a relative scale without the explosive number of comparisons that need to be done. Unfortunately this method looses the ability to build a complete ranking for all the clips. This method provides the performance of a a clip compared to a similarly ranked clip from the opposing system.

4.2.2 Data and selected clips

Both systems work with a written news article as the query input. Every news article has ten associated videos from which clips can be selected. These videos are preselected by the NISV prototype using metadata. The NISV prototype generates a jump-in point for each of these ten videos which will be used as the baseline in this experiment. The VCSS prototype analyzes the set of ten videos and selects a clip per video. These clips have a score towards matching the query and get re-ranked according to this score. So for each news article we have ten videos over which each system builds its own ranking. This process is illustrated in figure 4.3.

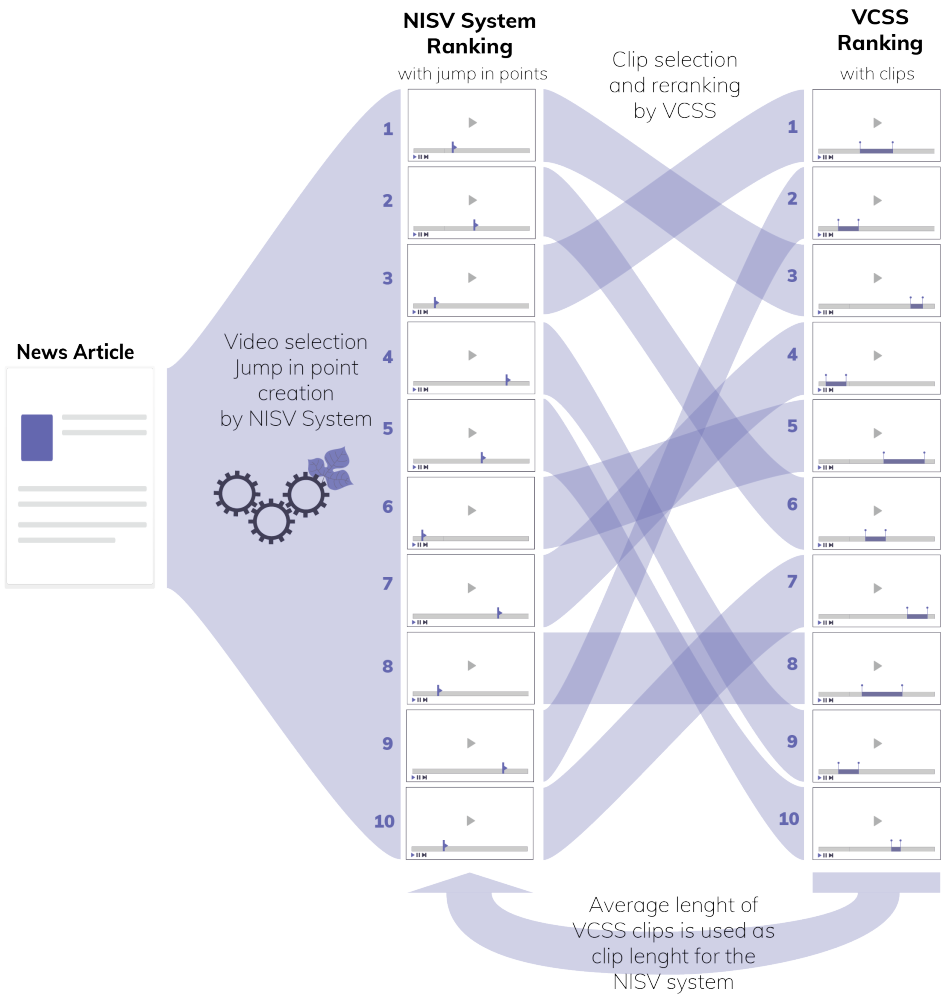


Figure 4.3: An article has 10 videos over which each system builds its own ranking.

The NISV prototype only gives jump points for the video. This is the point from which you should watch the video. The end of the clip is not specified. A stop point needs to be chosen to be able to call it a clip and compare it to the other system. A choice was made to take the average length of the VCSS system clips as a fixed length for the NISV system. This average is taken for each news article. This can be seen at the bottom of figure 4.3.

For each news article the top three ranked clips from each system are selected for the experiment. A set of three clips, one from each system and one random clip will be referred to as a comparison set. Three of these comparison sets are created per news article as can be seen in figure 4.4. The random clip is randomly selected from one of the videos that was pre-selected by the NISV for the news article. This clip starts from a random point in that video with a random duration between five and thirty seconds. This duration is within the possible length of a clip from the VCSS.

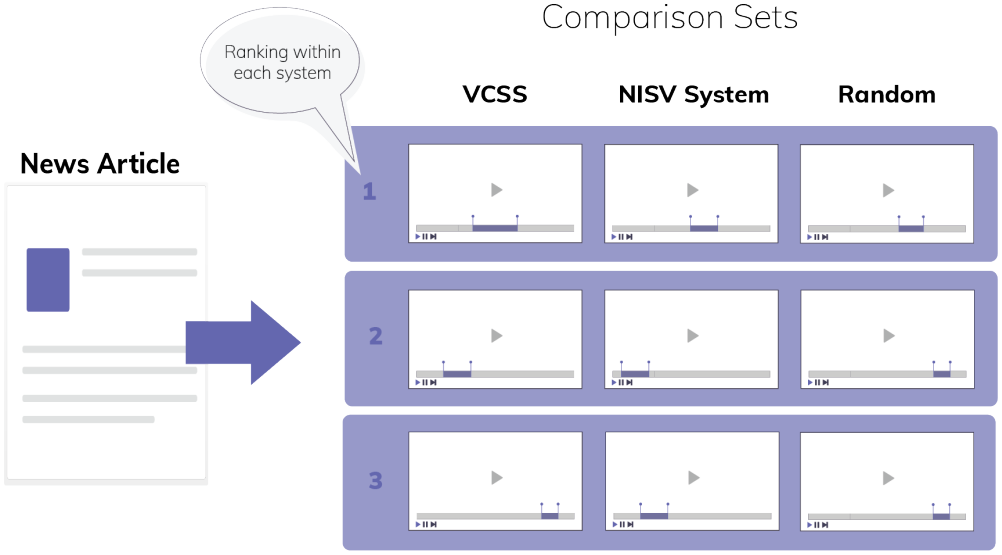


Figure 4.4: Each article produces three comparison sets.

4.2.3 Chosen news articles

Previous experiments showed that the performance correlates with the type of news articles put in to the proposed system. Articles on the economic or political side selected less visually interesting clips than articles that have an object associated with them like the technological articles which often feature an object. To test this theory and prevent this possible bias from influencing the outcome of the experiment a number of categories of news articles where chosen. The categories were chosen based on most common Dutch news categories.

- Culture and Media
- Economics
- Domestic news
- Foreign news
- Political
- Technology

The collected test-set that was used for the news articles did not have enough sports articles to include the category in this evaluation. This would make for an interesting news category since object detectors can properly detect sports attributes. There are more news categories out there, but the focus is on whether the news categories influence the performance, not which news categories ensures the most relevant video clips.

For each category two news articles were chosen from the NOS[1] a Dutch news organisation. These news articles were put through both systems and produced 108 clips within 36 comparison sets. An assessor can assess 36 comparison sets on two aspects.

4.2.4 Process enticement

Assessing 36 comparison sets on two aspects creates 72 assessment actions. Each clip that needs to be watched could be as long as 30 seconds. So 72 sets x three videos each x 30 seconds = 1,8 hours. To get to a compact assessment with a duration of less then an hour a steps were taken to shorten the amount of time that each assessor would need to spend on the experiment.

A decision was made to show each comparison set only once to the assessor. So if the assessor assessed a comparison set on the visual relevance aspect, the system would not ask the assessor to assess that particular comparison set on how interesting the clips are. This cuts the participation time in half. An positive side effect to this decision is that it avoids confusion. An assessor might get confused if he or she is asked to assess the same video clips multiple times.

When an assessor looses interest in the assessment process two things can happen. They stop the process and we loose all the data the assessor already assessed, since they did not complete the full process. Or they continue and pollute the data because they are trying to rush through. To prevent this potential data loss and pollution an participant is able to leave the experiment after assessing each comparison set. So an assessor can assess one comparison set or do them all. This could introduce some bias to the data since some assessors might not complete all the comparison sets. But we found the prevention of data loss and pollution a favourable trade off. To keep this bias to a minimum a global tally is kept for each comparison set. So the comparison sets that are assessed least are sent to new assessors first. This keeps the number of assessments per comparison sets equal.

The missing assessments from certain assessors might introduce a number of biases. Such as the learning effect [39] attention dropping over time and the fact that not every assessor completed the full experiment. That is why for each assessor the order of assessments is randomised: this is done for the order of the comparison sets, aspects and video clip presentation. The biases should have less influence with more assessors participating. The randomisation in combination with the volume of responses evens out the influence of the biases over the comparison sets.

[1] https://nos.nl/

With an estimated completion time of around an hour (reading articles etc. was not included in the previous calculation) and the notion that participants can stop at any time in the process we hope that the experiment is enticing enough to take part in.

4.2.5 Procedure

During the testing phase of this research the Covid-19 situation introduced itself to the world. This added a social distancing requirement to the testing procedure. An online assessment tool was created to enable people to assess the comparison sets from their own homes. This Evaluation tool will be described in section 4.2.6.

As described in the previous sections there are twelve articles with each three comparison sets that need to be assessed. A comparison set holds three video clips, one clip selected by the NISV system, one clip selected by the VCSS system and one randomly selected clip.

Two aspects need to be evaluated per comparison set, visual relevance and how visually interesting a clip is. When evaluating the clips certain conditions need to apply:

- To evaluate clips on visual relevance the context of the associated written article is needed. As the clip is evaluated on the visual relevance towards the article. So before a comparison set is evaluated on visual relevance the assessor has to have read the associated news article.
- When evaluating clips on how visually interesting they are, the context of a news article is not needed. In this experiment the news articles will not be shown when evaluating clips on this visual interest.
- Each comparison set is only shown to the assessor once, so either for visual relevance or interesting, but not both. Which clips are already assessed is tracked by the evaluation tool.
- Only the visual modality is evaluated, no audio will be provided during this experiment.
- To check the attention of the assessors and thus the ability to filter out polluted data, a set of questions is introduced which are asked after assessing clips on visual relevance. A detail is asked about the news article they just read. This is a simple check to see if they actually read the article.
- To make sure the clips are watched in their entirety the evaluation system wont let you advance before all the clips are watched. To encourage this, the clips are auto-played.

4.2.5.1 Procedural steps

The previously mentioned points result in the following assessment procedure:

1. *Introduction:* The assessor is asked to read instructions on how to assess properly and to read and sign a digital consent form. The assessor is made aware that he or she can stop the assessment process at any given moment.
2. *Selection:* The evaluations system selects a comparison set to evaluate and the aspect to evaluate on. If the selected aspect is visual interest, the comparison set is shown to the assessor directly. For visual relevance the assessor is presented with the news article associated with the comparison set to read.
3. *Clip presentation:* Each video clip from the comparison set is auto-played in succession in a randomised order. The assessor has the option to replay these clips as many times as they want.
4. *Assessment:* The user is asked to choose one of the three video clips that scores best on the aspect that they are currently assessing. The user sends their choice to the system.
5. *Validity check:* In the case of the visual relevance aspect the assessor is asked a fact about the previously read article to check if he or she actually read it.
6. *Repeat:* The process repeats from step two until all the comparison sets are assessed by this user, or the user presses the ”I want to quit” button.
7. *Final:* The user is thanked and given information on where to find the results of this experiment when they become available.

4.2.6 Evaluation Tool

This section will focus on the workings and interface of the digital evaluation tool. The actual experimental procedure is described in 4.2.5. The interface is presented in Dutch, since the experiment is conducted in Dutch.

4.2.6.1 Tasks

The evaluation tool a has a number of tasks it needs to perform. Keeping track of the user in the assessment process, keep track of the distribution of the amount of assessments per comparison set, present the user with comparison sets to assess and record the results.

Keep track of the assessment process of a user

Each assessment user is tracked through his or her process. This is done anonymously, so no personal data is gathered. A list is kept per user to track which comparison sets are already assessed by this user. This is done to be able to prevent a user from getting the same comparison set twice. A full list consist of the 36 comparison sets with two aspects, visual relevance and visual interest. As discussed in the previous sections a user only needs to complete one aspect per comparison set. So when a user completes one aspect for a comparison set the other aspect is removed as a possible assessment option for this user. When all the options are completed for a user, the evaluation system presents the ”Thank you” screen.

Keep track of the distribution of the number of assessments per comparison set

Each assessor can stop at any moment in the assessment process. This might leave certain comparison sets with less assessments when left to random chance. The evaluation system keeps track of how many times a comparison set is assessed. When a new assessor arrives, the comparison set with the least amount of assessments is presented first to that user. If there is more than one comparison set with the same amount of assessments a random comparison set is chosen from the ones with the minimal count. Over time this ensures an even distribution of assessments over all the comparison sets. The users that only partially complete the experiment will supplement each other.

Present user with comparison sets

When the user starts or when he or she completed the previous assessment the evaluation system sends a comparison set to the user until all the possible assessments are completed.

Record the results

The answers given by the users are collected centrally. For each user it is known which comparison sets they assessed and what answers they have given.

4.2.7 Interface

The interface is presented in the order a user would encounter them, with a small deviation. If a user gets an comparison set to assess on the *visual relevance* aspect or on the *how interesting* aspect is random. For presentation purpose the *how interesting* aspect is presented first followed by *visual relevance*.

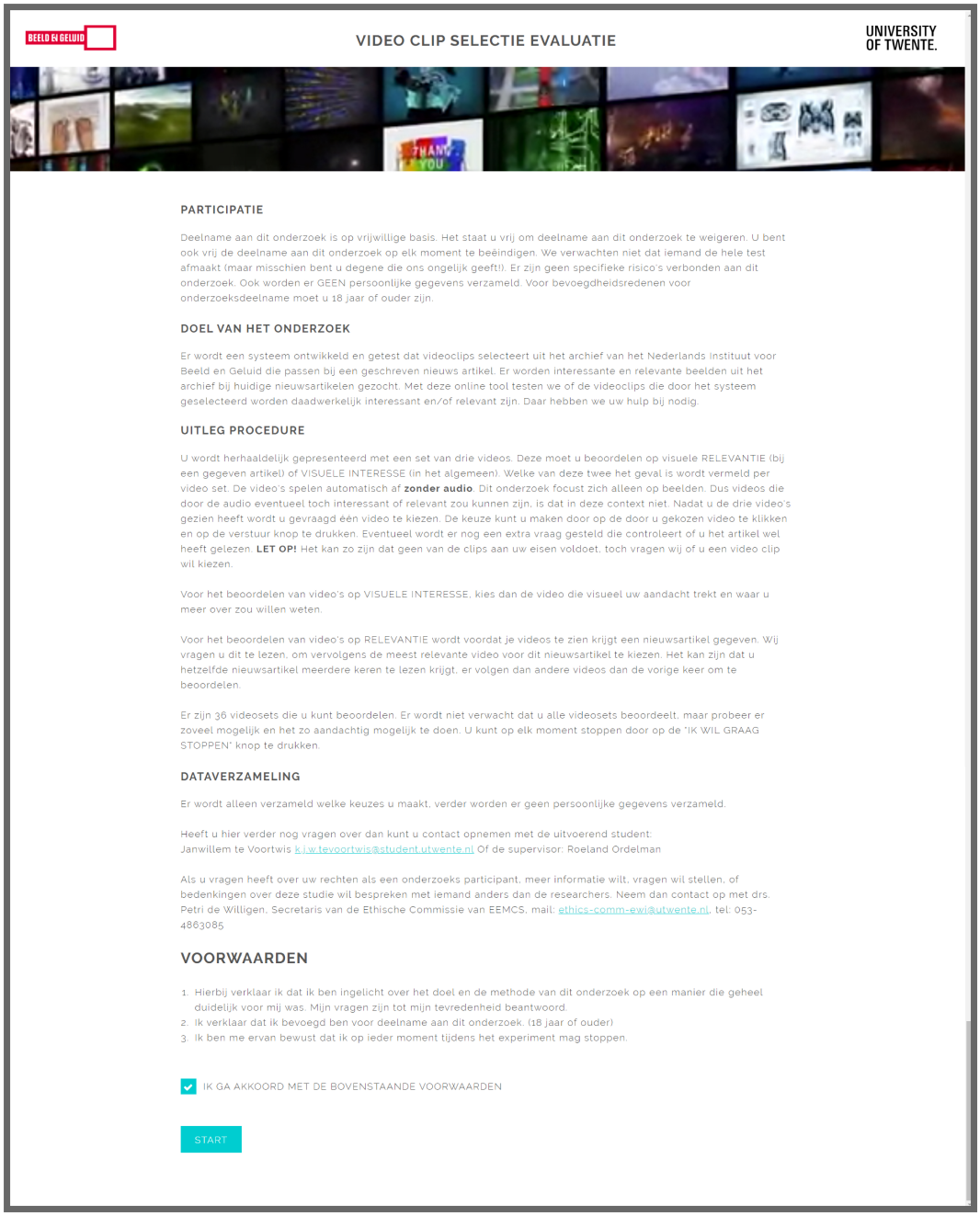


Figure 4.5: Instruction and Consent screen

A user logs in to the evaluation tool with a globally provided username and password and is presented with the explanation and consent screen as can be seen in figure 4.5. An explanation of the experiment is provided. The user is asked to sign a virtual consent form by opting a ”I agree” box.

After the user pressed start the system starts serving comparison sets. Each time a user encounters an aspect for the first time, a small explanation is given on how to assess for this aspect. For this presentation a comparison set that needs to be assessed on the *How visually interesting* aspect is given to the user first. The explanation for this aspect can be seen in figure 4.6.

Figure 4.6: Visually interesting instruction screen

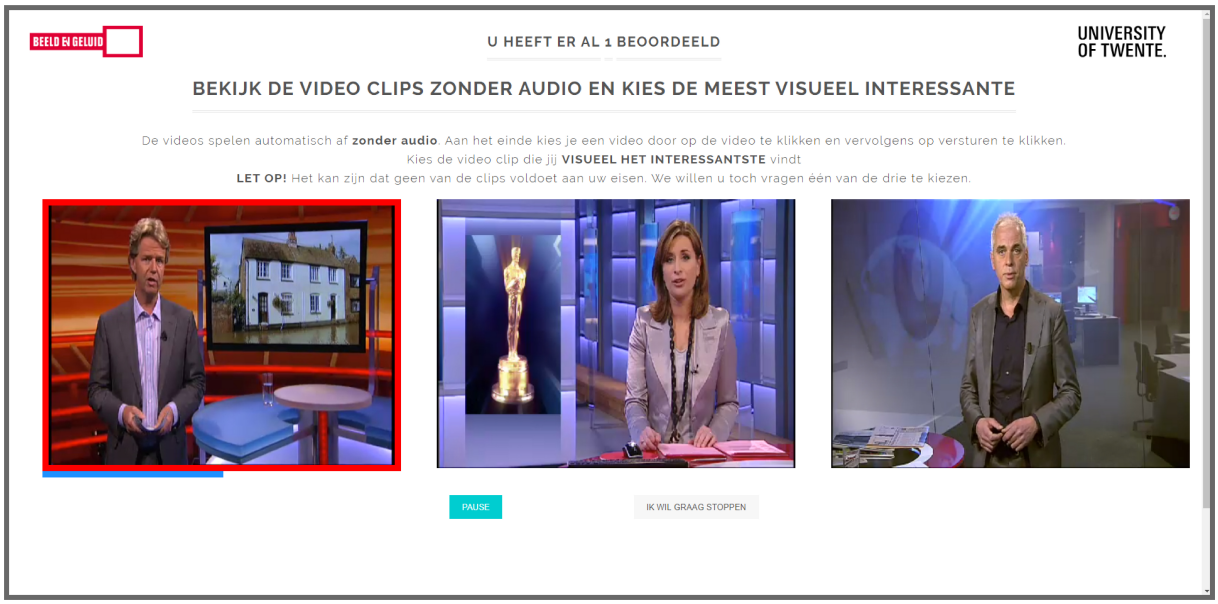


Figure 4.7: Video clip comparison screen - Start

,

When the user presses the button on the bottom of the page he or she is presented with the video clips as can be seen in figure 4.7. The top of the page lets the user know which aspect he or she is currently assessing. The video clips are automatically played after each other, from left to right. The currently playing clip has a red border around it to draw the attention of the user and a progress bar under it to indicate it is playing.

After all the clips finished playing you can replay them by clicking on their respective play buttons or the ”speel videos nog een keer” which replays all the videos. The assessor is asked to choose a video that best suits the aspect he or she is currently assessing. A video is selected by pressing the respective ”Kies deze” button or by a click on the videoclip. A green border appears around the chosen video clip as seen in figure 4.8 and the user can send the answer to the server. The user can also press the ”ik wil graag stoppen” button which takes the user to the thank you screen and stops the assessment process. For the presentation purposes we assume that the user presses the ”verstuur antwoord” button which sends their choice to the server.

For presentation purposes the user now encounters an assessment with the *visual relevance* aspect and is presented with the small instruction screen as can be seen in figure 4.9.

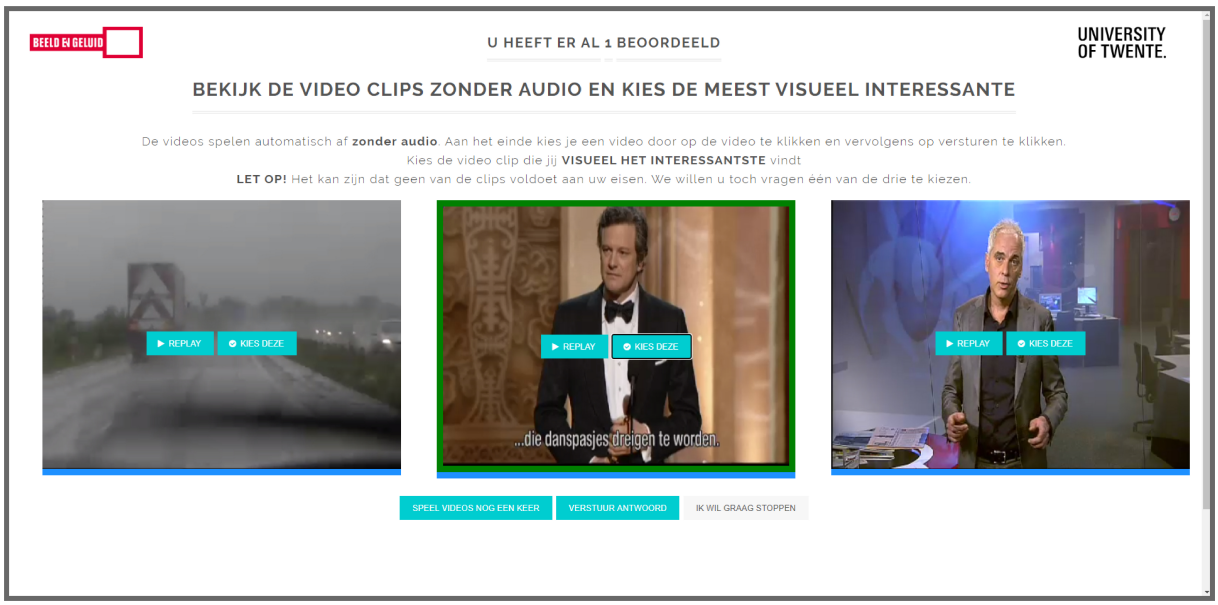


Figure 4.8: Video clip comparison screen - Choice

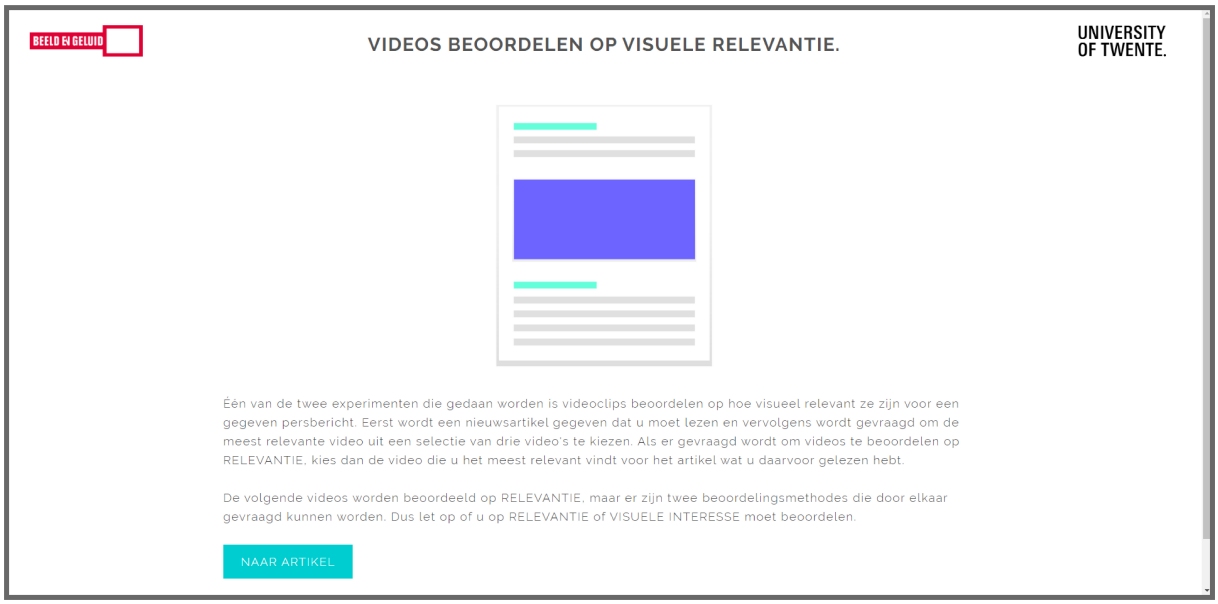


Figure 4.9: Visual relevance instruction screen

For each *visual relevance* assessment a news article associated with the comparison set is presented prior to the comparison screen. The user is asked to read the article and click the button on the bottom of the screen as presented in figure 4.10.

When the article is read the user is presented with the same comparison screen as before (figure 4.7), but with some differences. The text on the top represents the instructions for evaluating on the *visual relevance aspect*. And after a choice is made for a video clip an additional question appears that asks the user a question about the previous read news article. After a multiple choice answer is chosen the user can continue and choose one of the three previously discussed options at the bottom of the screen as illustrated in figure 4.11.

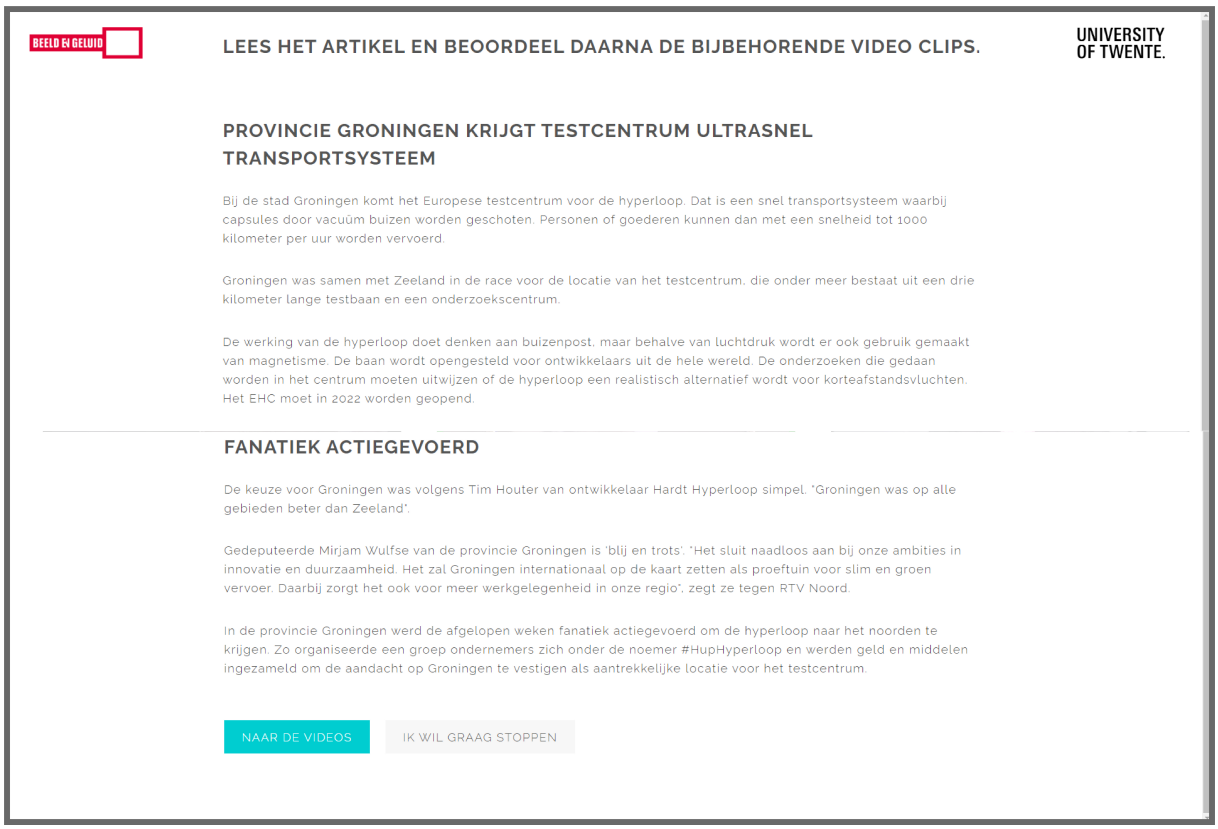


Figure 4.10: News article prior to visual relevance assessment

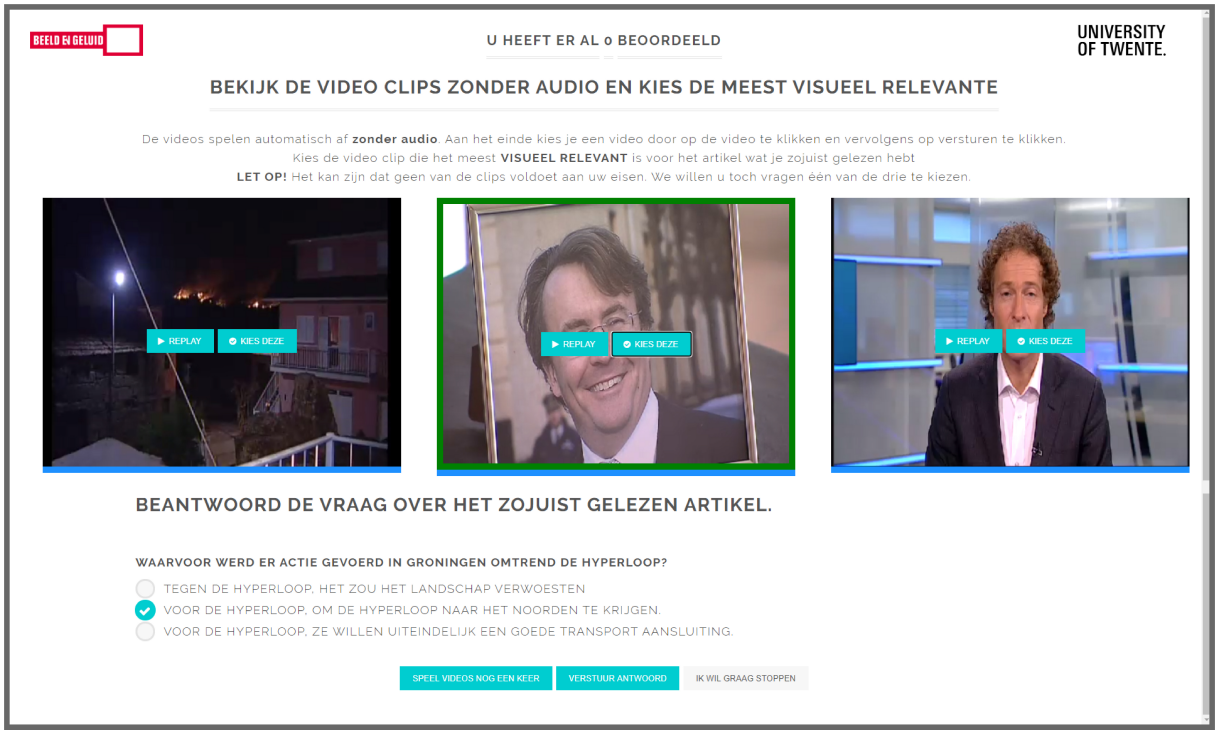


Figure 4.11: Video clip comparison screen - relevance

The process repeats until the user assesses all the comparison sets or clicked the stop button. Then the end screen appears and the users are thanked for their participation as can be seen in figure 4.12.

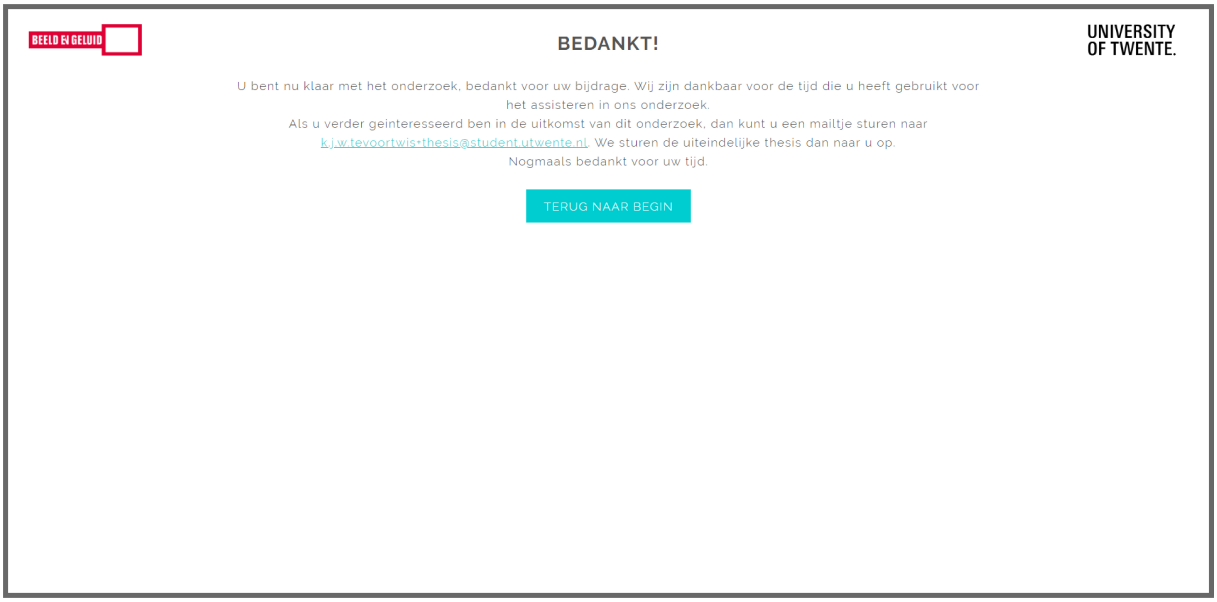


Figure 4.12: End and thank you screen

Chapter 5

Results & Discussion

5.1 Responses

The proposed and baseline systems were evaluated using an online evaluation tool, as described in section 4.2.6, that was accessible for two weeks. In this period fifty people took part. The participation was anonymous and the exact population parameters are not known, but they were recruited from employees of the Netherlands Institute for Sound and Vision and students from the University of Twente. As a consequence, we can expect a reasonable spread in the population in terms of age and background.

An average of 12 comparison sets were completed per participant. Four people completed more than 80% of the experiment. Most of the people completed less than a quarter of the comparisons. The distribution of how many people completed how many comparison sets can be seen in figure 5.1.

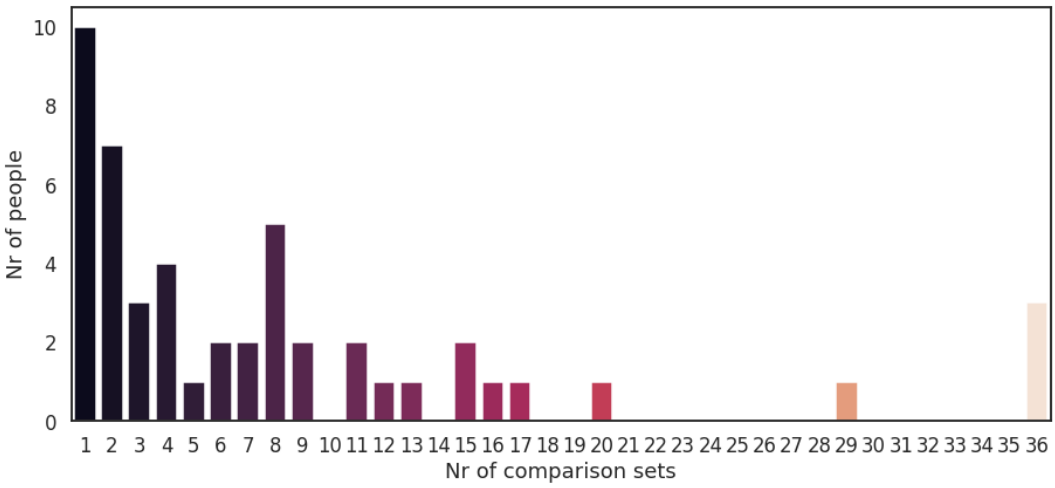


Figure 5.1: Number of people completing amount of comparison sets

Despite the relatively limited number of participants that completed the experiment, by pooling together all the answers every comparison set is annotated at least 4 times and at most 6 times, with a variance of 0.428. We assumed that many people do not want to spend an hour to complete the experiment. Looking at the graph this assumption was correct. To combat this the participants had the option to stop the experiment after completing a comparison set. Due to the COVID 19 situation the participants needed to do the experiment by themselves with their own motivation. The threshold to stop is much lower when there is no-one there that runs the experiment and looks over your shoulder.

5.2 Cumulative results

Three systems were compared to each other, the first system is the NISV prototype which is used as a baseline. The VCSS prototype, as described in chapter 3, will be referred to as the proposed system. A random system that selects random clips is referred to as the 'random' system.

The systems are compared on two aspects: the perceived visual relevance of the produced video clips ("Relevance") and how visually interesting those clips are perceived ("Interest"). In figure 5.2 the preference for each system is shown on these aspects. This is the cumulative data of all the comparison sets over all articles. Each bar represents the number of times that system was chosen as the most preferable. This is plotted for the "interest" and "relevance" aspects.

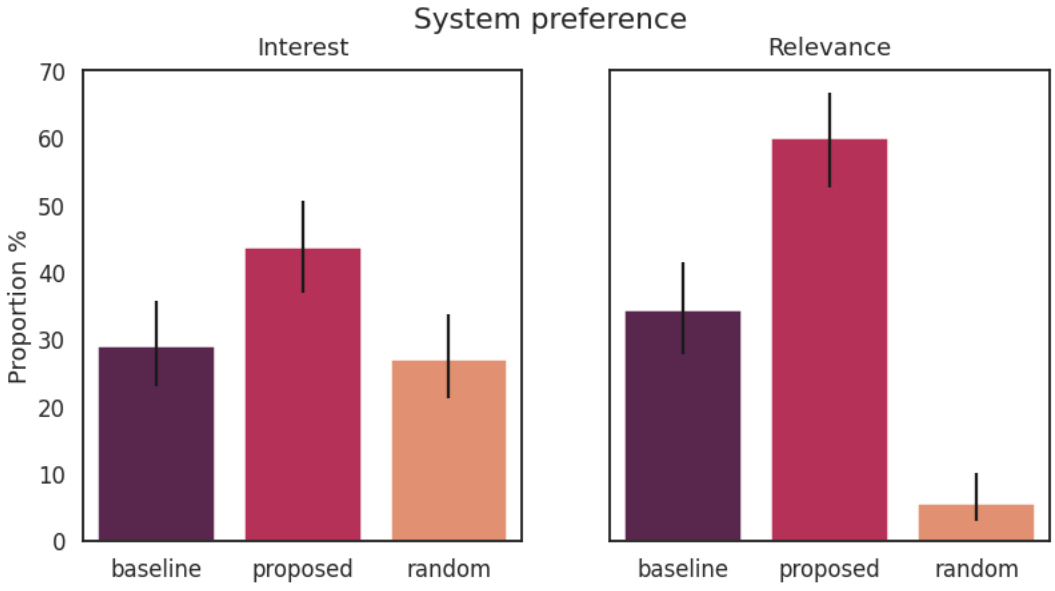


Figure 5.2: Preference for each system based on tested aspects

A 95% confidence interval is plotted on top of each bar to check if they overlap. The confidence intervals are calculated over the proportions using the Wilson procedure [40] with a correction for continuity. A chi-square goodness of fit test is performed with a null hypothesis that each system is equally preferred and an alternative hypothesis that at least one system is different. On both aspects the null hypothesis is rejected and we can assume that the systems perform significantly different. All significance testing is done using an hypothesis test for proportions with an 5% significance level.

On the ”interest” aspect, the baseline system did not outperform the random system significantly. This is proven by an hypothesis test with an H_0: baseline = random and a H_a: baseline $>$random. The hypothesis test produce a P-value which is the probability of obtaining test results at least as extreme as the results actually observed, under the assumption that the null hypothesis is correct. A very small p-value means that such an extreme observed outcome would be very unlikely under the null hypothesis. When lower than the significance level we reject the H_0 and accept H_a. The hypothesis test for baseline = random yielded a P value of 0.336 which is way higher than the significance level of 0.05 and does not reject the H_0. The baseline system not outperforming the random system impacts the performance of the proposed system. Where the baseline system can select videos from the entire NISV database the proposed system works with ten videos pre-selected by the baseline system. These ten videos are in this subset because they contain the clips selected by the baseline system. We assumed that this pre-selection contained interesting videos, but the baseline selections not performing better than a random system questions this. While this does not impacts the direct comparison between systems, it does suggest that the performance for the proposed system is underestimated since it worked of a sub-set that holds videos that might not be that visually interesting.

The proposed system outperformed the baseline system on the ”interest” aspect significantly with the hypothesis test yielding a P value of 0.0006. This is lower than the significance level resulting in rejecting the H_0: proposed = baseline and accepting the H_a that the proposed system is significantly preferred over the baseline.

When looking at the ”relevance” aspect the results are much clearer. The proposed system is preferred over the baseline system and the baseline is preferred over the random system. The baseline system performs better than random on the ”relevance” aspect, which does not question the assumption that the pre-selected videos by the baseline systems are relevant. While this assumption was not tested, we believe that a better pre-selection will improve the performance of the proposed system.

5.3 Filter results

A check was introduced to filter out respondents who were not paying attention when annotating the ”relevance” aspect. An easy question was asked about the article that they should have read. The responses with the incorrect answer were then filtered out. Figure 5.3 shows the relevant aspect data with and without these responses filtered.

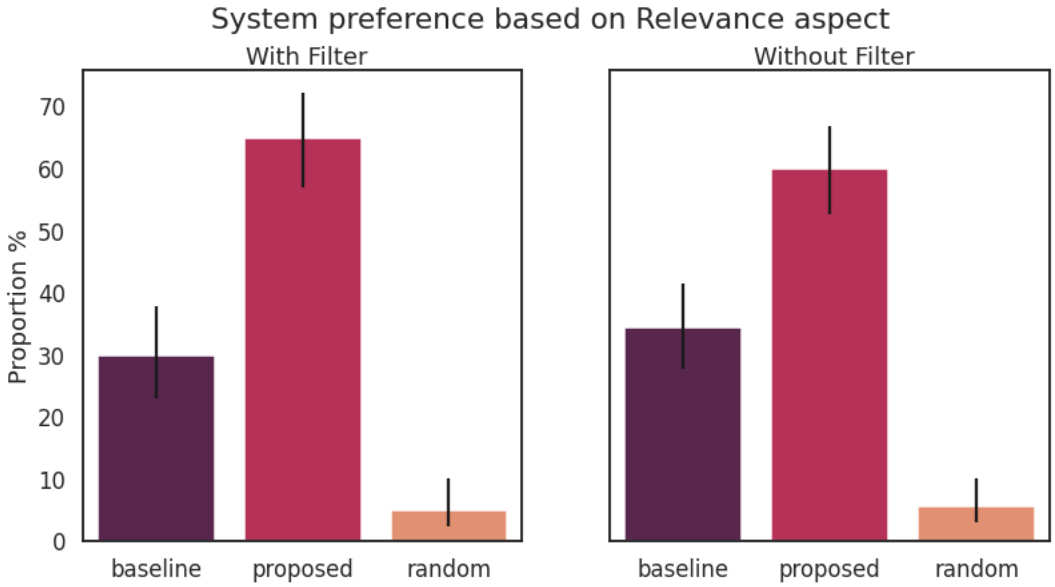


Figure 5.3: Relevant aspect preference with and without attention filter

A number of responses get filtered out, but there is not a big shift in preference outcome for ”relevance”. The ratio between the systems does change slightly. The baseline system loses 4.4%, the random system loses 0.5%. This shifted to the proposed system which gained 4.9%. Passing this check suggest that the proposed system is not simply enticing the participants in choosing for the proposed system because of better visual interest.

5.4 Article Categories

The systems are evaluated using news articles from six different news categories with the assumption that the categories have influence on the performance of the new system. The assumption was that more object related news articles, for example the technology and culture category, would outperform the more topic related categories like politics and economics on the ”interest” aspect. The system preference on both aspects grouped per news category can be seen in figures 5.4 and

5.5.

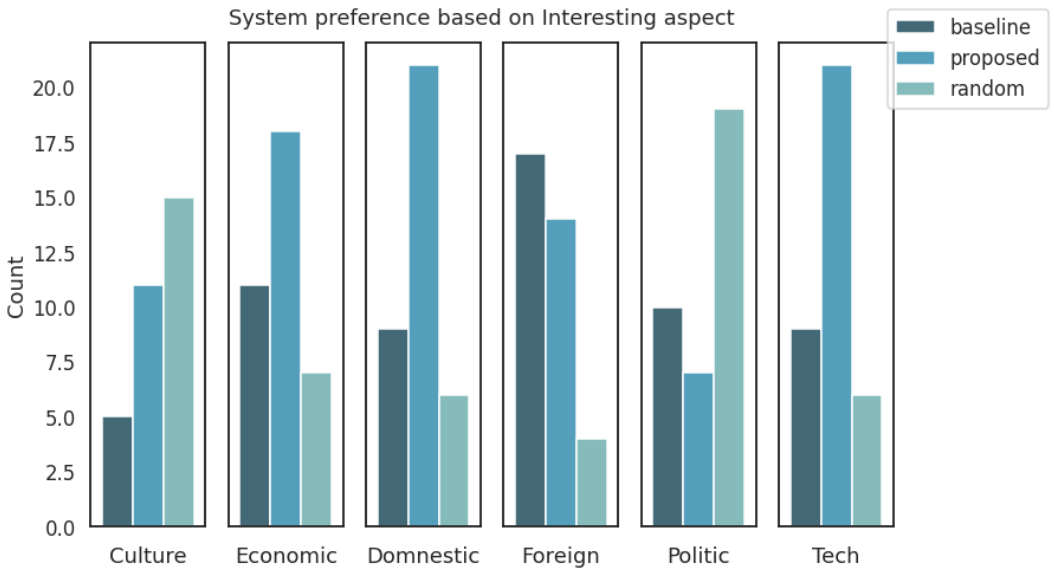


Figure 5.4: Preference for each system per news category for interesting aspect

The proposed system performs poorly in the political category on both aspects as was expected, but outperforms the other systems in almost all other categories. When looking at the culture category we see that the proposed system selected relevant clips, but these clips were not very visually interesting since the clips from the random system are chosen more often. This could be because the subjects of the articles in the culture category are not that visually interesting. In this case the first article is about plagiarism of a book and the second one is about who won the golden globes. Another possible explanation can be the higher number of talking heads that are present in the clips from the baseline and proposed systems in this category. Talking heads are people that are mostly static, like most news anchors. An example can be seen in figure 5.6. Talking heads are most of the time not that interesting to look at.

During the creation of the evaluation clips it became clear that there are a number of near duplicates in the pre-selected videos. For example, two news broadcast where the news and the clips are exactly the same, but only the anchor is different. The proposed system chooses only one clip from a video to combat near duplicate clips. But when two videos are nearly the same, the proposed system can select near duplicate clips which is not favourable. This resulted in two near duplicate clips that ended up in the evaluation process. With two out of 72 clips compromised (2.7%) this was accepted as a flaw in the process since it was discovered so late. These clips in question were associated with the news article about the Oscars in the culture category.

In the foreign category the baseline system is preferred on the ”interest” aspect. The proposed system is very close. Both articles in this category produce very visually busy clips. The first article is about a man killed by the police and produces mostly clips of people rioting. The second article is about the leave of an Iranian general which produces clips of very big masses of people. The random system selected a number of talking heads and a number of environment shots. The political category shows that the clips that are relevant to the news articles are not very interesting. The random system outperformed the other systems with clips that had nothing to do with the articles. This suggests that the political category does not produce very interesting clips, more or less the same as the culture category.

The technology and domestic categories have news articles that were very object focused. The tech articles were about the hyper-loop and a spaceship. The domestic articles were about a crashed windmill and a demonstration with torches. We see that for the proposed system these more object related articles result in high preference on the ”interest” aspect as was expected.

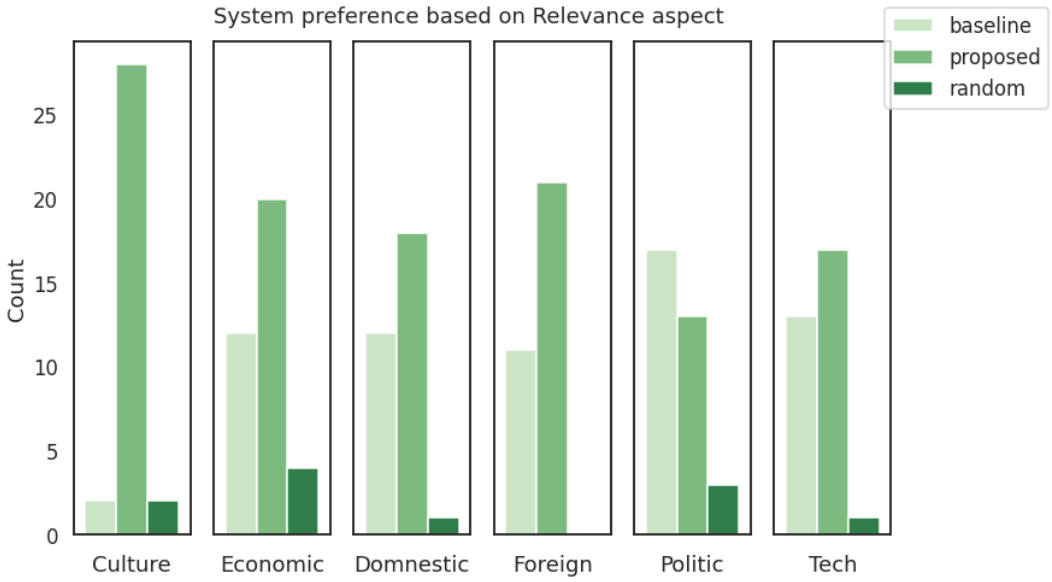


Figure 5.5: Preference for each system per news category for relevance aspect

The ”relevance” aspect shows that both systems are capable in selecting visually relevant clips since the random system is not preferred very often. The proposed system outperforms the baseline on all categories except politics. An outlier is the culture category where almost everyone preferred the proposed system. The proposed system mainly selected clips where an object or environment is shown that is relevant to the article. The baseline system produces mainly talking heads and the random system produced clips that had nothing to do with the articles. People preferred the related objects and environments over the talking heads. On the more object related news articles from the technology and domestic categories we see

Figure 5.6: An example of a talking head

that the proposed system selected clips with a relevant object or environment and the baseline selected mostly talking heads. It is the same observation as in the culture category, but we do not see a clear preference or the proposed system such as in the culture category in the tech en domestic categories. The talking heads are not completely visually irrelevant.

These graphs indicate that the news category has an influence on the performance of the systems. As this experiment was done using two articles per category, there is not enough data to draw a definitive conclusion about the influence of different news categories.

5.5 System Ranking

Clips produced by the old and new system have a ranking associated with them. In this experiment the clips with the same ranks were compared to each other. The first rank would have the clips that are best suited to the news article according to the system, the second rank the second best, and so on. The system preference on both aspects based grouped by system rank can be seen in figures 5.7 and 5.8.

Looking at the ”interest” aspect the proposed system outperforms the other systems on the second and third rank, but not significantly better than random on the first rank with a p value of 0.4247. On the first rank the baseline system is significantly better than the proposed system with a p value of 0.0384. The baseline system does not outperform the random system on the second and third rank. An explanation for the baseline outperforming the proposed system on only the first rank

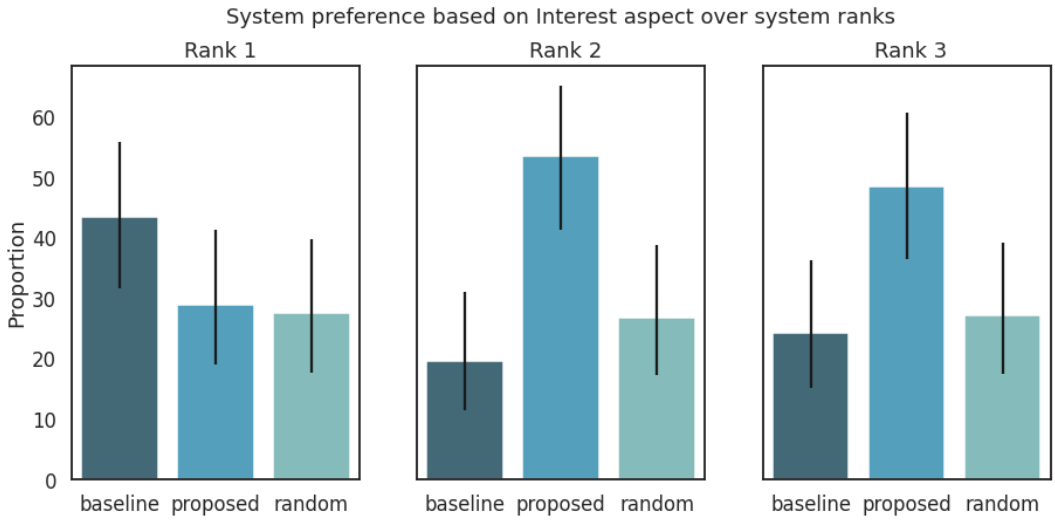


Figure 5.7: Preference for each system per system rank for interest aspect

can be that the baseline system focuses on performance on the first rank where the proposed system is more geared towards performance over more ranks. The random system has a relative high preference on the ”interest” aspect when compared to the ”relevance” aspect. This either shows that both systems on some level do not select visually interesting video clips or that the annotation process of the ”interest” aspect is more ambiguously spreading out the choice.

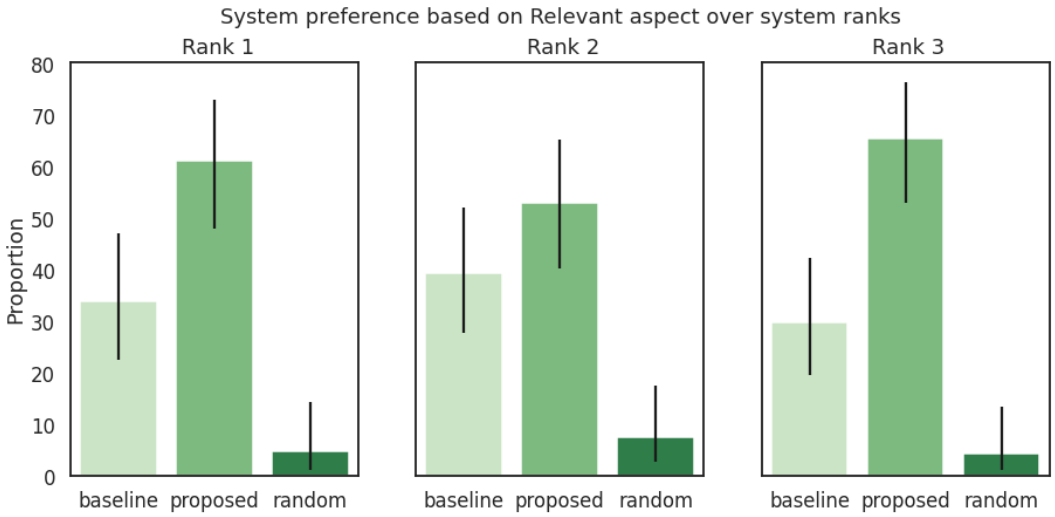


Figure 5.8: Preference for each system per system rank for relevance aspect

The proposed system does outperform the other systems on the first and third rank looking at the ”relevance” aspect. On the second rank it comes close, the proposed system does not significantly outperform the baseline system with an p value of 0.0582. There is no obvious trend over the ranks on the ”relevance” aspect. On the second rank the proposed system loses in preference, giving to both the

random and the baseline system compared to the other ranks. This change is not at a level that suggests a difference between ranks.

5.6 Observations during experiment

A number of observations were made during the development of the proposed prototype or during the evaluation that might influence or bias the results of the experiment.

Sometimes the baseline system produced impossible jump in points. For example the jump in point would be set at an hour when the video was only half an hour long. The articles associated with these clips were not included in the experiment where possible. But with a scarcity of testable news articles one such case ended up in the experiment. In this case the clip with the jump in point error was ignored and the clip from the next rank was used instead.

A big part of the available videos are news broadcasts. This type of content contains a lot of talking heads. With talking heads being not that visually interesting, the systems may work better on other types of content. It would be interesting to see the system tested on a wider range of content.

The approach of connecting semantic terms to visually detected objects through term mining was abandoned because it showed some problems. The main problem was that more general objects like ”person” have many of associated terms. One way we might solve this, but needs more thorough investigation, is applying a statistical test to the term-concept matching to filter out the important terms. Automating the term-concept matching through term mining makes the usage of object detectors with much higher concept counts a lot easier, since you do not have to make these matches manually.

The novelty of the field of query dependent clip selection posed a challenge for the evaluation of the proposed system. A big aid for further research would be the creation of an standardised evaluation dataset. This evaluation dataset holds a reference standard that results from proposed systems can be tested against. An evaluation data set allows for easy testing, less resources needed to evaluate a model and shortens the evaluation process. During the design process of the proposed system, a number of assumptions needed to be made because we did not have the capacity to evaluate all the options. We assumed that a broad concept detector improves the clip selection more than an specialised concept detector. Another assumption was that object detection was favourable over object classification. With the evaluation dataset these assumed options can be tested easily against their counterparts.

Chapter 6

Conclusion

This research reports on the evaluation of including the visual modality in a query dependent clip selection process. The selected videoclips are evaluated on the aspects of visual relevance and how visually interesting the selections are. This was done using a proposed system that brings the visual and speech modality to the same semantic Word2Vec feature space for an easy comparison to the query.

The overall improved performance of the proposed system over the baseline shows validity for bringing the detected objects and speech data to the same semantic Word2Vec feature space. Using this method the objects and speech data are easily fused and compared to the query. This provides an relative accessible platform to do query dependent clip selection that includes the visual modality, since it only uses pre-trained models.

Our results show that including the visual modality in the clip selection process significantly improves both the perceived visual relevance and perceived visual interest of the selections. The clips that were selected by the proposed system were preferred over a random selection and clips selected by a baseline system that did not use this modality. The study suggests that the type of query has an influence on the performance of the proposed system. The results reflected that ”object-related” queries performed better than the ”topic related” queries for the proposed system. For example, a query about a rocket would be more ”object-related” and a query about a political standpoint would be more ”topic related”. To confirm that these observations are indeed structural further research is needed.

The baseline system does not perform significantly better than a random clip selection when evaluating visual interest. The proposed system does beat both random and baseline, but not as convincingly as when evaluating on visual relevance. This shows there is still progress to be made, especially on the aspect of visual interest. Improving further on selecting visually interesting videoclips is important, because it helps systems with catching the attention of users. Providing visually relevant clips helps systems with improving the perceived quality of the selections. We

believe that the visual modality holds the information to improve query dependent clip selection even further.

6.1 Future work

Looking at our results, talking heads seem to have a negative impact on the visual interest of video clips. It would be interesting to detect these shots with talking heads and incorporate this feature in the clip selection process. By penalising shots with talking heads the visual relevance and visual interest of the selection might benefit.

This research used public broadcast content that contained mostly news broadcasts. Future work should verify these conclusions by running the experiment on different types of content. Films, web and social videos are a big part of consumed visual media and it would be interesting to see if the proposed query dependent clip selection method works on these types of content.

A big aid for further research would be the creation of an standardised evaluation dataset. An dataset with a reference standard that methods can be tested against instead of using human assessors would speed up the evaluation process and make it less resource intensive. The creation of this reference standard is nontrivial. The documents may contain only parts that are relevant to a certain query, so standard relevance and precision measures would not be that useful since they work on a document level. A measure using overlap of the selected clips with annotated relevant parts in the document is then a more useful measure. This annotation can be done for relevance, given a query what parts of the video a relevant. For interest this would be more ambiguous.

The selected clips of the proposed system are directly derived from the segments. It would be interesting to explore incorporating the shot boundaries back in to the final selection by looking at nearby boundaries to the start and end of the segment. Setting the start and stop points to shot boundaries can create more logical start and stop points of the clips. A requirement for this is that the shot boundary information is available or it can be extracted from the video. The clips that have a start and stop points at shot boundaries can then be evaluated against the original clip to see if the perceived quality, relevance and interest changes.

This research proved that bringing all the modalities to a single semantic feature space is viable for query dependent clip selection. It would be interesting to expand on the Word2Vec fusion approach. This research focused on the visual and speech modality. Other modalities and metadata can be added to improve the clip selection process. Other concept detectors can be tested instead of object detection. There are many options to expand and improve this proposed method.

References

[1] Z. Elkhattabi, Y. Tabii, and A. Benkaddour, "Video summarization: Techniques and applications," *International Journal of Computer and Information Engineering*, vol. 9, no. 4, pp. 928 – 933, 2015. [Online]. Available: https://publications.waset.org/vol/100

[2] B. Xiong, Y. Kalantidis, D. Ghadiyaram, and K. Grauman, "Less is more: Learning highlight detection from video duration," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019.

[3] A. G. Hauptmann and M. H. Smith, "Text, speech, and vision for video segmentation: The informediatm project," 1995.

[4] T.-S. Chua and L.-Q. Ruan, "A video retrieval and sequencing system," *ACM Trans. Inf. Syst.*, vol. 13, no. 4, pp. 373–407, Oct. 1995. [Online]. Available: http://doi.acm.org/10.1145/211430.211431

[5] T. Mei, Y. Rui, S. Li, and Q. Tian, "Multimedia search reranking: A literature survey," *ACM Comput. Surv.*, vol. 46, no. 3, pp. 38:1–38:38, Jan. 2014. [Online]. Available: http://doi.acm.org/10.1145/2536798

[6] H. D. Wactlar, T. Kanade, M. A. Smith, and S. M. Stevens, "Intelligent access to digital video: Informedia project," *Computer*, vol. 29, no. 5, pp. 46–52, 1996.

[7] A. G. Hauptmann, "Lessons for the future from a decade of informedia video analysis research," in *Image and Video Retrieval*, W.-K. Leow, M. S. Lew, T.-S. Chua, W.-Y. Ma, L. Chaisorn, and E. M. Bakker, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2005, pp. 1–10.

[8] C. G. M. Snoek and M. Worring, "Concept-based video retrieval," *Foundations and Trends in Information Retrieval*, vol. 2, no. 4, pp. 215–322, 2009. [Online]. Available: http://dx.doi.org/10.1561/1500000014

[9] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016.

[10] M. Sang, Z. Sun, and K. Jia, “Semantic similarity based video reranking,” in *2015 International Conference on Computational Intelligence and Communication Networks (CICN)*, Dec 2015, pp. 1420–1423.

[11] A. P. Natsev, A. Haubold, J. Tešić, L. Xie, and R. Yan, “Semantic concept-based query expansion and re-ranking for multimedia retrieval,” in *Proceedings of the 15th ACM International Conference on Multimedia*, ser. MM ’07. New York, NY, USA: ACM, 2007, pp. 991–1000. [Online]. Available: http://doi.acm.org/10.1145/1291233.1291448

[12] G. A. Miller, R. Beckwith, C. Fellbaum, D. Gross, and K. J. Miller, “Introduction to WordNet: An On-line Lexical Database*,” *International Journal of Lexicography*, vol. 3, no. 4, pp. 235–244, 12 1990. [Online]. Available: https://doi.org/10.1093/ijl/3.4.235

[13] M. Sun, K. Zeng, Y. Lin, and A. Farhadi, “Semantic highlight retrieval and term prediction,” *IEEE Transactions on Image Processing*, vol. 26, no. 7, pp. 3303–3316, July 2017.

[14] B. Xiong, Y. Kalantidis, D. Ghadiyaram, and K. Grauman, “Less is more: Learning highlight detection from video duration,” *CoRR*, 2019. [Online]. Available: http://arxiv.org/abs/1903.00859

[15] K. Niu and H. Wang, “Video highlight extraction via content-aware deep transfer,” *Multimedia Tools and Applications*, vol. 78, no. 15, pp. 21 133–21 144, Aug 2019. [Online]. Available: https://doi.org/10.1007/s11042-019-7442-6

[16] T. Mikolov, K. Chen, G. S. Corrado, and J. Dean, “Efficient estimation of word representations in vector space,” *CoRR*, 2013.

[17] T. Mikolov, I. Sutskever, K. Chen, G. Corrado, and J. Dean, “Distributed representations of words and phrases and their compositionality,” *CoRR*, 2013. [Online]. Available: http://arxiv.org/abs/1310.4546

[18] D. Cer, Y. Yang, S. Kong, N. Hua, N. Limtiaco, R. S. John, N. Constant, M. Guajardo-Cespedes, S. Yuan, C. Tar, Y. Sung, B. Strope, and R. Kurzweil, “Universal sentence encoder,” *CoRR*, 2018. [Online]. Available: http://arxiv.org/abs/1803.11175

[19] Chong-Wah Ngo, Ting-Chuen Pong, and R. T. Chin, “Video partitioning by temporal slice coherency,” *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 11, no. 8, pp. 941–953, Aug 2001.

[20] E. Apostolidis and V. Mezaris, “Fast shot segmentation combining global and local visual descriptors,” in *2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, May 2014, pp. 6583–6587.

[21] S. Tippaya, S. Sitjongsataporn, T. Tan, M. M. Khan, and K. Chamnongthai, “Multi-modal visual features-based video shot boundary detection,” *IEEE Access*, vol. 5, pp. 12 563–12 575, 2017.

[22] H. Bay, T. Tuytelaars, and L. Van Gool, “Surf: Speeded up robust features,” in *Computer Vision – ECCV 2006*, A. Leonardis, H. Bischof, and A. Pinz, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2006, pp. 404–417.

[23] J.-M. Odobez, D. Gatica-Perez, and M. Guillemot, “On Spectral Methods and the Structuring of Home Videos,” IDIAP, Martigny, Switzerland, Idiap-RR Idiap-RR-55-2002, 2002, published in International Conference on Image and Video Retrieval, 2003.

[24] L. Baraldi, C. Grana, and R. Cucchiara, “Shot and scene detection via hierarchical clustering for re-using broadcast video,” in *CAIP*, 2015.

[25] L. Baraldi, C. Grana, and R. Cucchiara, “Scene segmentation using temporal clustering for accessing and re-using broadcast video,” in *2015 IEEE International Conference on Multimedia and Expo (ICME)*, June 2015, pp. 1–6.

[26] D. Rotman, D. Porat, and G. Ashour, “Robust and efficient video scene detection using optimal sequential grouping,” in *2016 IEEE International Symposium on Multimedia (ISM)*, Dec 2016, pp. 275–280.

[27] T. G. A. Smith and N. Pincever, “Parsing movies in context,” in *USENIX Summer*, 1991.

[28] J. Huang, V. Rathod, C. Sun, M. Zhu, A. Korattikara, A. Fathi, I. Fischer, Z. Wojna, Y. Song, S. Guadarrama, and K. Murphy, “Speed/accuracy trade-offs for modern convolutional object detectors,” *CoRR*, 2016. [Online]. Available: http://arxiv.org/abs/1611.10012

[29] A. Kuznetsova, H. Rom, N. Alldrin, J. Uijlings, I. Krasin, J. Pont-Tuset, S. Kamali, S. Popov, M. Malloci, A. Kolesnikov, T. Duerig, and V. Ferrari, “The open images dataset v4: Unified image classification, object detection, and visual relationship detection at scale,” *IJCV*, 2020.

[30] x. tian and D. Tao, “Visual reranking: From objectives to strategies,” *IEEE MultiMedia*, vol. 18, no. 3, pp. 12–21, March 2011.

[31] Y. Song, M. Redi, J. Vallmitjana, and A. Jaimes, “To click or not to click: Automatic selection of beautiful thumbnails from videos,” *CoRR*, 2016. [Online]. Available: http://arxiv.org/abs/1609.01388

[32] J. Beel, B. Gipp, S. Langer, and C. Breitinger, “Research-paper recommender systems : a literature survey,” *International Journal on Digital Libraries*, vol. 17, no. 4, pp. 305–338, 2016.

[33] C. De Boom, S. Van Canneyt, T. Demeester, and B. Dhoedt, “Representation learning for very short texts using weighted word embedding aggregation,” *Pattern Recognition Letters*, vol. 80, pp. 150 – 156, 2016. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0167865516301362

[34] P. K. Atrey, M. A. Hossain, A. El Saddik, and M. S. Kankanhalli, “Multimodal fusion for multimedia analysis: a survey,” *Multimedia Systems*, vol. 16, no. 6, pp. 345–379, Nov 2010. [Online]. Available: https://doi.org/10.1007/s00530-010-0182-0

[35] J. Korhonen, N. Burini, J. You, and E. Nadernejad, “How to evaluate objective video quality metrics reliably,” in *2012 Fourth International Workshop on Quality of Multimedia Experience*, July 2012, pp. 57–62.

[36] Y. Baveye, E. Dellandréa, C. Chamaret, and L. Chen, “From crowdsourced rankings to affective ratings,” in *2014 IEEE International Conference on Multimedia and Expo Workshops (ICMEW)*, July 2014, pp. 1–6.

[37] E. Parizet, E. Guyader, and V. Nosulenko, “Analysis of car door closing sound quality,” *Applied Acoustics*, vol. 69, no. 1, pp. 12 – 22, 2008. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0003682X06001939

[38] G. N. Yannakakis and J. Hallam, “Ranking vs. preference: A comparative study of self-reporting,” in *Affective Computing and Intelligent Interaction*, S. D’Mello, A. Graesser, B. Schuller, and J.-C. Martin, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2011, pp. 437–446.

[39] P. Linares, S. Lumbreras, A. Santamaría, and A. Veiga, “How relevant is the lack of reciprocity in pairwise comparisons? An experiment with AHP,” *Annals of Operations Research*, vol. 245, no. 1, pp. 227–244, October 2016. [Online]. Available: https://ideas.repec.org/a/spr/annopr/v245y2016i1d10.1007_s10479-014-1767-3.html

[40] R. G. Newcombe, “Two-sided confidence intervals for the single proportion: comparison of seven methods,” *Statistics in*

Medicine, vol. 17, no. 8, pp. 857–872, 1998. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1002/%28SICI%291097-0258%2819980430%2917%3A8%3C857%3A%3AAID-SIM777%3E3.0.CO%3B2-E