Students and declaration

|  |  |
| --- | --- |
| **Student Name 1:** | Santiago de Arribas de Rendo |
| **Student ID Number 1:** | 16449272 |
| **Student Email 1:** | santiago.dearribasderenedo2@mail.dcu.ie |
| **Student Name 2:** | Ayman El Gendy |
| **Student ID Number 2:** | 15395461 |
| **Student Email 2:** | ayman.elgendy2@mail.dcu.ie |
| **Student Name 3:** | Fawaz Alsafadi |
| **Student ID Number 3:** | 15380871 |
| **Student Email 3:** | fawaz.alsafadi2@mail.dcu.ie |
| **Student Name 4:** | Stephen Stephanov |
| **Student ID Number 4:** | 16372836 |
| **Student Email 4:** | stephen.stephanov2@amil.dcu.ie |

**DCU School of Computing Assignment Submission**

Student Name(s): Fawaz Alsafadi, Santiago de Arribas de Renedo, Stephen Stephanov, Ayman El Gendy

Student Number(s): 15380871

Programme: BSc in Computer Applications

Project Title: Search technologies Research and Development Project

Module code: CA4009

Lecturer: Gareth Jones

Project Due Date: 17/12/2019

**Declaration**

I declare that this material, which I now submit for assessment, is entirely my own work and has not been taken from the work of others, save and to the extent that such work has been cited and acknowledged within the text of my work. I understand that plagiarism, collusion, and copying is a grave and serious offence in the university and accept the penalties that would be imposed should I engage in plagiarism, collusion, or copying. I have read and understood the Assignment Regulations set out in the module documentation. I have identified and included the source of all facts, ideas, opinions, viewpoints of others in the assignment references. Direct quotations from books, journal articles, internet sources, module text, or any other source whatsoever are acknowledged and the source cited are identified in the assignment references. I have not copied or paraphrased an extract of any length from any source without identifying the source and using quotation marks as appropriate. Any images, audio recordings, video or other materials have likewise been originated and produced by me or are fully acknowledged and identified. This assignment, or any part of it, has not been previously submitted by me or any other person for assessment on this or any other course of study. I have read and understood the referencing guidelines found at http://www.library.dcu.ie/citing&refguide08.pdf and/or recommended in the assignment guidelines. I understand that I may be required to discuss with the module lecturer/s the contents of this submission. I/me/my incorporates we/us/our in the case of group work, which is signed by all of us.

**Signed:** Fawaz Alsafadi, Santiago de Arribas de Renedo, Stephen Stephanov, Ayman El Gendy

Contents

[Abstract 3](#_Toc27213948)

[Example User interface 3](#_Toc27213949)

[Introduction 4](#_Toc27213950)

[User Analysis 4](#_Toc27213951)

[1. The user 4](#_Toc27213952)

[2. Using the system 4](#_Toc27213953)

[2.1 Search queries 4](#_Toc27213954)

[2.2 Search result 5](#_Toc27213955)

[3. Operational scenarios 5](#_Toc27213956)

[Scientific Functional Description 7](#_Toc27213957)

[1. Overall architecture 7](#_Toc27213958)

[1.1 Algorithms 7](#_Toc27213959)

[1.2 Design 8](#_Toc27213960)

[Evaluation plan 9](#_Toc27213961)

[4.1 Evaluation strategy 9](#_Toc27213962)

[Here are some examples of how it would work: 9](#_Toc27213963)

[4.2 Overall objective 9](#_Toc27213964)

[4.3 Data used 9](#_Toc27213965)

[4.4 Search queries 9](#_Toc27213966)

[4.5 Collection of data 10](#_Toc27213967)

[4.6 Evaluation metrics 10](#_Toc27213968)

[Accuracy and Precision 10](#_Toc27213969)

[Examples include 10](#_Toc27213970)

[Conclusion 10](#_Toc27213971)

[References 11](#_Toc27213972)

Video Summarization

# Abstract

This report details the implementation of a search system for videos. It uses video summarisation as a method to break down footage into labelled sections relating to different events or objects. This process uses audio analysis and image processing via computer vision. A user’s query is then matched to each of these sections and, if successful, these sections are relevant for what the user is searching for.

The system will be run as a web application and the user will input a query and supply a video as a YouTube link or in a recognisable format. E.g. .MP4. The system is assuming that the user can input a query that relates to what they *want* to search for and that they provide a video of good quality - high video resolution and high-quality audio.

The back-end of the system will use a server to send the users query request and video to be processed. The system will then return the processed data which can then be displayed on the front-end of the web application. This processed data will be returned as footage in a timeline format and point the user to sections in this footage that match their query.

# Example User interface

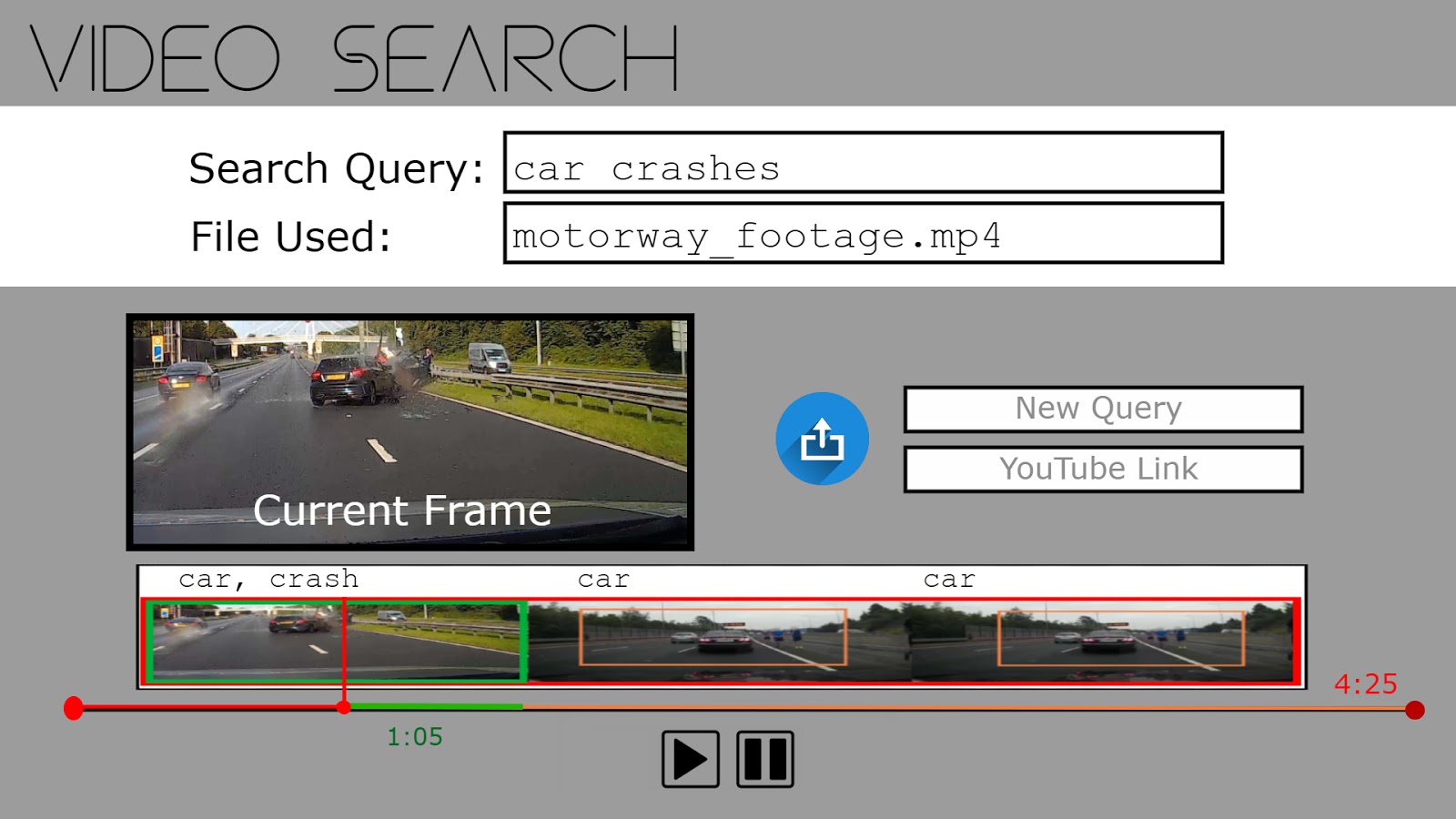


Fig 1.0 Web Application interface concept of Video Search system

# Introduction

As video is becoming more and more prevalent online, it is essential that we find ways to search through all this generated content. Video can come from CCTV footage, conferences, lectures and a plethora of other sources. Since conventional Information Retrieval (IR) systems work on the basis on pulling relevant documents to satisfy a user’s query, it is evident that these will not work for this new emerging media. Our system hopes to build a solution that will allow users to satisfy these queries.

According to *medium.com*, video summarisation “is the process of distilling a raw video into a more compact form without losing much information.” This system aims to use video summarisation as a component in developing a feature that will enable a user to search through the content of a video file. Portions of the video will be summarised based on their two major factors: their **audio transcripts**, which have been processed through semantics analysis, and the **images**, which have been processed via computer vision.

There a few systems on the market which perform similar functionality to what this system hopes to achieve. However, they do not seem to implement the use of video summarisation. An example of this would be the *TalkSearch* software. This allows users to search through conferences and talks by analysing the audio transcripts and matching these with what a user queries.

A system close to what we aim to build would be Microsoft's *Video Indexer* which uses audio transcripts, facial recognition and image recognition to allow users to search for specific elements in the videos they are looking for. Analysing how Video Indexer works is useful in determining what current solutions are working which can influence the development of this system.

The system should break down a video into different sections, each section using semantic analysis and computer vision to summarise what the content of the section is about. A user will input a query of an element they wish to search in the overall video and will be matched with the appropriate section.

# User Analysis

## 1. The user

While designing this system one of our main goals was to make sure that it is as accessible to the widest range of users as possible. The target user base for this application is any private individual using this application in order to find an object or an event in a video. The user will not need to have specialised knowledge to use this system, the user simply needs to navigate to our system which is in the form of a Web Application in their preferred browser.

We believe removing the need for any specialized or technical knowledge will allow the widest possible range of users from all age to user our application. The application will then navigate the user to the point in the video which they were searching for, building the system with minimal interaction requirements will allow for much improved accessibility.

## 2. Using the system

### 2.1 Search queries

In order to execute a valid search, the user will have to provide two items a *search query* and a *video*.

* *Search query:* The search query will be an object the user is hoping to find in a video, for example if the user is watching a documentary about whales, the user can provide the word “whale” in order to find all the times a whale appears in the video.

Furthermore, the user is also able to provide an event as a query, this will expand the use of the system to be able to find more than just objects and identify events during the video. This will be accomplished using smart context detection. An example event a user can search for is if they are looking to find all the times a person has slipped in a collection of clips of people slipping on ice. Natural language processing will extract the meaning from a query such as “slipping on ice” and using smart context detection the application will be able to return time stamp off all the times a fall occurs in the video.

* *Video:* Along with a valid search query the user must also provide a video to be searched. The video can be supplied through a link to a video streaming service such as “YouTube” or uploaded in “.mp4” format.

### 2.2 Search result

Once the search has been successfully executed the user will then be redirected to their video in the browser. The video will begin playing at the timestamp of the first occurrence of the search object or event.

In the case that the search query appears multiple times during the video the user will also be provided with an interactive list of timestamps which the user can user to navigate to other sections of the video where their search query appears.

## 3. Operational scenarios

From the perspective of the user the system is designed to be as simple and straightforward as possible. As such we will be focusing on a single operational scenario. This scenario will begin with a user navigating to the Web Application and here the user will supply the system with a search query (An object or event) and a video they would like to perform a search on.

Once the search has been executed the front end will then contact the backend API’s and supply the search query and video. Once processing completes the user will then be shown the video starting at the first instance where the object or event occurs; with timestamps to other parts of the video if the object or event occurs multiple times.

The following diagrams will help to visualize and explain this scenario.

* *Sequence diagram* – Highlights the sequence of steps between user entering search query to when the user receives a result.
* *Search operation flow –* Flow of steps from when the user navigates to the Web application to when the user has used the system successfully and exits the application.

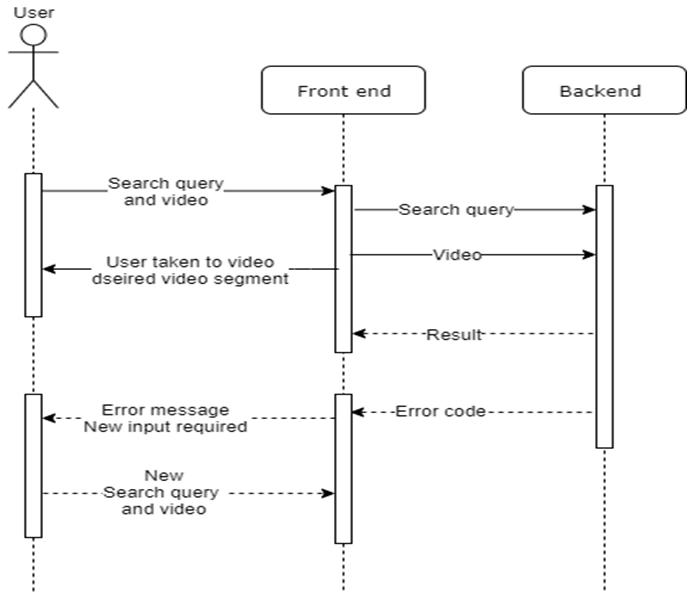


Figure 3.1 - Sequence diagram

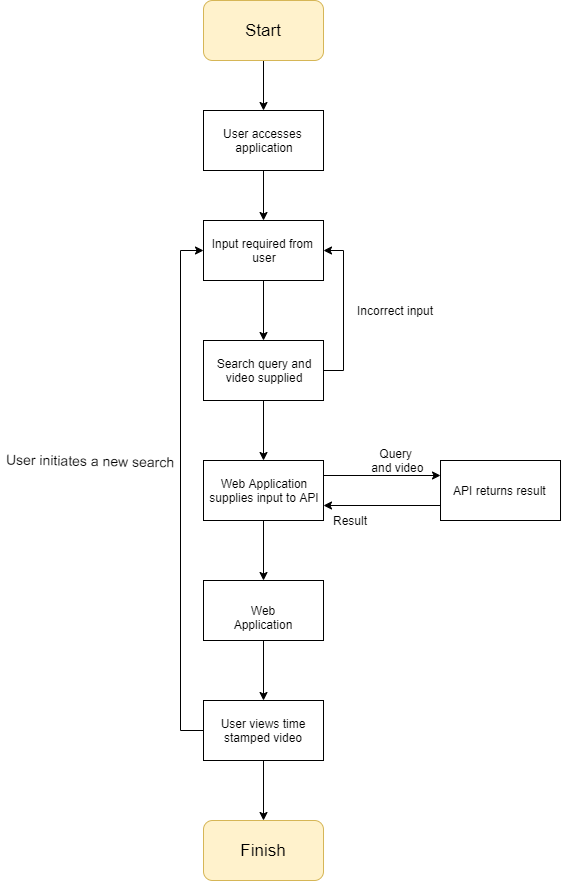


Figure 3.2 - Search operation flow

# Scientific Functional Description

## 1. Overall architecture

The systems entire architecture is comprised in various sub-components and comprised of different phases for algorithmic computation and rule detection using neural networks. The overall physical design is made up of a front-end and back-end system. The client-side will be a web application which will be the user’s tool or GUI in accessing the systems functionality. From here the user submits a video through the web app alongside the action they wish to be found inside the video. The information is sent to the server which provides the computation for processing said video. Algorithms and methods are invoked on the video to understand the computers request programmatically and the result is re-submitted to the front-end application.

After computation, the query of the user is referenced for the desired action/scene to be retrieved from the video. If successful, the server responds with a video summary of the actions sought by the user in the same compressed format. The user can then analyse and compare the initial video pre-processing versus the retrieved result showcasing minified content strips in a timeline format represented in the user interface. This is the procedure for a working scenario. In a case where the system may fail to produce a result, the system presents the various scenarios that may be related to the users query on the initial video timeline. This however is not a tackled issue and assumptions will be made of the users intended results which may or may not be what they’d be anticipating.

### 1.1 Algorithms

The system will incorporate a variety of algorithms to perform data analysis to assist in creating a video summarisation. The areas in which the algorithms will be used to summarise video data is in motion, colour and audio features.

#### 1.1.1 Motion

Tracking algorithms are used in measuring current positions of objects in a given physical representation by means to predict their next whereabouts in future frames.

Point tracking decides which points are best to track relative their local image contrast. For each point that is successfully tracked, a vector corresponding to the displacement between its frames is computed.

Points that cannot be successfully matched are dropped and the points retrieved are subject to a limitation of points to be computed and minimum distance between one another. The known difficulties with this algorithm are that highly textured images tend to generate many points.

#### 1.1.2 Audio

A clustering algorithm here is used to group small relevant audio segments in the form of ‘claps’. The caveat here in using such methods with audio is to be able to determine a detection rule for audio. Reasons for this may be to detect different sounds or even overlapping sounds such as clinking cutlery, glasses or even a car door slamming.

Such rules could be implemented through the assistance of support vector machines, trained neural network models or even gradient/grid search for the detection of grid parameters.

#### 1.1.3 Colour

The idea behind colours in video summarisation arises from being able to combine the information alongside motion to determine “interesting” shots.

Here, the same computations such as distance is used for colour features and a distribution matrix is generated. A temporal term is used alongside segment computation to penalize segments in the matrix that are far apart in time, this ensures redundant shots aren’t prevalent.

### 1.2 Design

#### 1.2.1 Data

As previously described, the user will be accessing the system through a front-end interface and supply a video of their choice with the action they wish to find contained inside. The video provided into the pass-through interface should be of a recognised format/ frequently used format that can be recognised.

There is no limit as to how long the video may be as an upload, however it would be made clear to the user that having excessive lengths processed by the system will not incur fast processing speeds.

#### 1.2.2 Hardware

Most algorithms undergo matrix computation and as a result, the requirement for building a commercially available system is quite high. This is due to the natural cost of GPU’s which are essentially used for number crunching amongst various algorithms and artificial intelligence techniques.

For the cost of the system to be effective, there must be a balance between performance and the budget of computer hardware where one factor does not overrule or predetermine the other factor due to mostly expenditure constraints.

#### 1.2.3 Design

It is assumed for the majority of the time; the user shall receive their expected video summarisation result upon their video being completely processed. The system is dependent on several factors:

Firstly, the user must have given clear instructions to the system about the actions they specified. It is entirely up to the user to input this semantically correct.

Secondly, the video footage should be in a reasonably good resolution. The footage can display several different qualities in which the video can be computationally summarised. The potential in the systems power lies in containing clustering algorithms and supervised learning algorithms. The clustering algorithms join segments of ‘interesting’ footage based on inner content. The supervised artificial intelligence models attempt to identify the scene/action in question for the user, using previous footage knowledge. Both these techniques are combined in hope to achieve a quality video summarisation.

Lastly, the data models being utilised must hold a wide generic acceptance criterion for unforeseen data (i.e. must not suffer from overfitting). Having fulfilled these prerequisites, the user should receive summarised video footage to hold their descriptions intent.

# Evaluation plan

## 4.1 Evaluation strategy

The main approach we are going to use in evaluating our system will be to compare all our generated summaries to a set of test/ reference summaries in our data set. Furthermore, we will compare computer generated summaries to human generated summaries to compare the accuracy of the system. The overall purpose of these comparisons is to determine how accurately the system can get in relation to the user’s needs.

### Here are some examples of how it would work:

*Search query*: Dog, House, Garden

*Expected result:*Video jumps to position that is best fits the search query. (e.g. Dog inside house, Dog in garden, Dog in front of house in garden, House and its Garden).

Our strategy will be to get users to test this expectation using search queries for videos they are familiar with and seeing whether it takes them to the position in the video that they were expecting to reach.

## 4.2 Overall objective

The overall objective of our evaluation strategy is to make sure our system works as close to perfect as possible, according to the software specification.

During the evaluation stage the system should identify any errors or problems that would make the software less than optimal. Examples include, but are not limited to the following:

1. The search query jumps to a completely irrelevant position in the video, our evaluation should determine this as the software failing.
2. The search query locates a relevant location in the video but according to the user’s previous knowledge of the video, there is a more relevant location for the video to jump to. This would not be considered a complete failure in our evaluation; however, it does point towards a less efficient ranking system which can be improved for better user satisfaction.

Evaluating the system by following the above format should at least result in a working software that can retrieve relevant search results in video summarization.

## 4.3 Data used

The data used for the evaluation stage will be a large collection of videos that vary quite differently in their nature/category. The reason for this is that we want to explore the functionality using a large range of relevant and irrelevant search queries. The accuracy of a video exploring some nature setting should be just as reliable as the accuracy in a video containing an interview or speech. Some of the videos in the test collection will be pre-selected videos and some will be random videos selected by the test users.

## 4.4 Search queries

Search queries should follow one of the following formats:

1. Queries that consist of descriptive comma separated words that can be used to search the video accurately. For example, “Man, Stage, Microphone”.
2. Short sentences that work in a similar fashion to a google search for example “Man gives speech at conference stage”.

Search queries should not contain long paragraphs that make it difficult to identify the key attributes of the search.

## 4.5 Collection of data

Data will be collected using testing. Testing will occur in three stages:

1. Testing done by the development team: We will test the software using videos and queries chosen by us and we will collect the data from the results manually.
2. User testing: People who agree to use the software will perform their own search queries on videos of their choice. We will collect the data from their search results and whether the search queries met the user’s expectations.
3. We will use randomly generated search queries and random videos to collect data on the systems functionality, especially when search queries don’t match any video segments.

## 4.6 Evaluation metrics

### Accuracy and Precision

* *Accuracy* is straight forward, it will be the number search queries that were successful over the number of total search queries tested. If the accuracy is less than 80%, it should be too low by the systems standards and users expectations.
* *Precision* will be the portion of search queries that deemed successful and retrieved that are related to the search query.
* *Recall* will be the portion of search queries that were successful over the portion of search queries that should have been successful:

### Examples include

Search queries = “Road, Ambulance” “Car, Collision”, “Man, Angry, “Dog, Cat”. If the video has a segment where there is a car crash and an ambulance, three out of four of the search queries should be successful. If only 2 are successful, then our recall is 2/4.

If all four of the search queries retrieve the same video segment, then our precision is 3/4. As only three of the four selected segments were correct

# Conclusion

In conclusion, the system will be a web application that allows the user to submit a query and a video file/ YouTube link to a video they wish to search through. Users are expected to have little to no technical experience and should be able to intuitively use the system.

The system will use a simple client-server architecture to send the video data from the front-end of the web application to the back-end server. This server will then use external algorithms to process the video by analysing the audio to perform semantic analysis and computer vision to process the visuals.

We will evaluate the system by using a specific set of queries and videos that are known to contain sections that match those queries. A set of videos that do not match the queries will also be used. This should give us good testing data to work with.

# References

* Surya Remanan (2019) *Video Summarization: A Brief Introduction,* Available at: *https://medium.com/dataseries/video-summarization-a-brief-introduction-30ee0dbbb008* (Accessed: 11th November 2019).
* (2019) *55 video marketing statistics for 2019,* Available at: *https://biteable.com/blog/tips/video-marketing-statistics/* (Accessed: 11th November 2019).
* Adam Hayes (2019) *The State of Video Marketing in 2019 [New Data],* Available at: *https://blog.hubspot.com/marketing/state-of-video-marketing-new-data* (Accessed: 11th November 2019).
* Francine ChenMatthew CooperJohn Adcock (2007) *System and method for video summarization Abstract,* Available at: *https://patents.google.com/patent/US8200063B2/en* (Accessed: 12th November 2019).
* Nancy Messieh (2017) *Did You Know You Can Search Within a YouTube Video?,* Available at: *https://www.makeuseof.com/tag/search-within-youtube-video/* (Accessed: 13th November 2019).
* Dave W. Shanahan (2016) *Microsoft Garage releases “Video Breakdown,” generates searchable transcripts and a lot more,* Available at: *https://www.onmsft.com/news/microsoft-garage-releases-video-breakdown-generates-searchable-transcripts-and-a-lot-more* (Accessed: 13th November 2019).