Simão Pedro Nogueira Carvalho

Licenciado em Engenharia Informática

Merging Augmented Reality with Television Shows to Enhance the Viewer Experience

Dissertação para obtenção do Grau de Mestre em

Engenharia Informática

Orientador:	Professora Doutora Teresa Isabel Lopes Romão Faculdade de Ciências e Tecnologia Universidade Nova de Lisboa
Co-orientadores:	Doutor Pedro Miguel da Fonseca Centieiro Viva Superstars Digital Media, Lda

Júri:

Presidente:	Professor Doutor Miguel Pessoa Monteiro
Vogais:	Professora Doutora Ana Paula Pereira Afonso Professora Doutora Teresa Isabel Lopes Romão

FACULDADE DE CIÊNCIAS E TECNOLOGIA
UNIVERSIDADE NOVA DE LISBOA

Setembro, 2018

Merging Augmented Reality with Television Shows to Enhance the Viewer Experience

Acknowledgements

The realization of this thesis would not be possible without the help of my advisor Professor Dr. Teresa Romão, and my co-advisor Dr. Pedro Centieiro. I wanted to thank them for helping me choose the theme and solving some technical problems.

I would also like to thank Viva Superstars for all the equipment they provided me during this project and all their members who were always ready to help in whatever needed.

I also want to thank my family and friends, especially my grandparents. Without them, none of this would have been possible.

Abstract

Nowadays, television no longer has the same effect on viewers as it had decades ago. The “traditional” television has been losing audience over the years in favor of new technologies. The time that was formerly spent watching television, was replaced by smartphones and tablets, where the viewer has the opportunity to interact with the content that is provided to him, receiving stimuli that television cannot offer on its own. More and more people are looking for new ways to socialize and interact outside the space they are confined to, in order to discuss certain topics and watch videos, or images published by others. This makes the concept of watching television, just for the pleasure of watching, an old-fashioned concept that needs to be adapted to the modern times. This thesis aims to introduce innovative concepts of interactivity in television contexts, and to achieve it, we will explore the possibility of integrating augmented reality (AR) concepts with television shows to enhance the viewer experience. By using AR, we can view objects and information that otherwise would not be possible, simply because they do not exist in our reality or the original movie. This technology is earning an important role in our day-to-day activities, namely in the entertainment area. Our goal is to allow viewers to watch and interact with TV shows through a mobile device and use AR elements to present important information and amusing effects by overlaying the video content. With this approach, we hope to introduce a new way of interacting with TV shows so that we can meet the expectations of a new generation of audiences. Taking into account the results we had, this concept can be considered a success and can possibly be one of the next steps in TV show user interaction.

Keywords: Augmented reality, television, social, interaction, mobile devices, entertainment.

Resumo

Hoje em dia, a televisão já não tem o mesmo efeito nos telespectadores como tinha há décadas atrás. A televisão "tradicional" tem vindo a perder audiência ao longo dos anos a favor das novas tecnologias. O tempo que antes era passado a ver televisão, foi substituído por smartphones e tablets, onde o espectador tem a oportunidade de interagir com o conteúdo que lhe é fornecido, recebendo estímulos que a televisão não pode oferecer. Cada vez mais as pessoas procuram novas formas de socializar e interagir fora do espaço a que estão confinados, para discutir certos tópicos e assistir a vídeos ou imagens publicadas por outras pessoas. Isso torna o conceito de assistir televisão, apenas pelo prazer de assistir, um conceito antiquado que precisa ser adaptado aos tempos modernos. Esta tese tem como objetivo introduzir conceitos inovadores de interatividade em contextos de televisão e, para alcançá-lo, será explorada a possibilidade de integrar conceitos de realidade aumentada (RA) com programas de televisão para melhorar a experiência do espectador. Ao usar RA, podemos visualizar objetos e informações que, de outra forma, não seriam possíveis, simplesmente porque não existem na nossa realidade. Esta tecnologia é relativamente nova, mas está a ganhar um papel importante nas nossas atividades do dia-a-dia, nomeadamente na área de entretenimento. O nosso objetivo é permitir que os espectadores assistam e interajam com programas de TV através de um dispositivo móvel, podendo recorrer a elementos de RA para visualizar informações importantes e efeitos divertidos que sobrepõem o conteúdo do vídeo. Com esta abordagem, esperamos apresentar uma nova maneira de interagir com os programas de TV, para podermos corresponder às expectativas de uma nova geração de audiências. Tendo em conta os resultados que obtivemos, este conceito pode ser considerado um sucesso e possivelmente pode ser um dos próximos passos na interação entre utilizadores e programas televisivos.

Palavras-Chave: Realidade aumentada, televisão, social, interação, dispositivos móveis, entretenimento.

Contents

List of Figures

List of Tables

1

Introduction

Since television became available to the world's population, it became an essential part of our daily routine. Nowadays, television as we know it, has a major role in modern society. This technology can show us news from around the world and it can be a reliable mean of entertainment. That made television the main mean of socializing and seeking information [28] until recently, when some changes in this scenario started to happen [81]. One of the main reasons for the television popularity is the quantity of entertainment shows that viewers have available to them at any instant.

Today, to answer the necessities of a new generation of audience, we can find a wide range of choices, from action movies to reality shows, so it is relatively easy to please all viewers [29,112]. This was great when television had no competition. Now we can find more than one technological device in our homes. Almost everyone possesses a smartphone or a tablet and uses it while watching television, and in most cases, this use is not related with what the user is watching on television [21]. If we want to socialize, we no longer need to share the same physical space, we only need a mobile device, or a computer connected to the Internet. On top of that, young generations spend more time on their smartphones than watching TV, caring less with it [81]. Since smartphones and tablets can do the same as traditional television and much more, television is losing the audience that used to have among the younger generations.

With the evolution of technology, this new generation of audience, needs new kinds of entertainment. They live in a world far more social than their ancestors. Television needs to turn into a more interactive medium of socialization. From the creation of television to the present day, this technology has suffered some changes, some evolutions to adapt to the new generations of viewers (such as the introduction of TV set boxes and IPTV). It has become possible to watch a show outside the time of its broadcast, either because you want to watch it later or because the broadcast has already begun, and you want to see it from the beginning. One can also record a certain show, or even a whole season, so that it can be seen later. If a user is not near a television, he can also watch his shows from a device with access to the Internet. In line with these technological advances in the TV industry, we seek to push the boundaries of what is possible to do with the content that we are watching, particularly when watching a TV show on a mobile device.

1.1. Motivation and context

Young generations usually watch television while using their smartphones, either to search information, sometimes about what they are watching, or just for interacting in a social network [65]. The number of people who owns a mobile phone, worldwide, is exorbitant. It is estimated that more than 63% of world's population is in possession of such a device, where half of them possess a smartphone, and that by 2019, the number of smartphone users will rise to almost 3 billion [78].

Humanity, by nature, is a social species. People have the need to comment on something they like and discuss topics of their interest [83]. We use social networks to socialize with all kinds of persons (friends or people we may never have met in real life), to search for information, to spend time when we are bored, or simply for our personal pleasure [102]. Thanks to this kind of social behaviors, a new type of hobby has risen. It is very usual to use our mobile devices to access the internet and consume different forms of entertainment. For example, someone can take a snapshot of a certain situation that happened in a television show and alter its content, either by adding text or some graphical element. The purpose is to post it online so other users can see it and provide comic relief. One good example of this is shown in Figure 1.1, taken from [56].

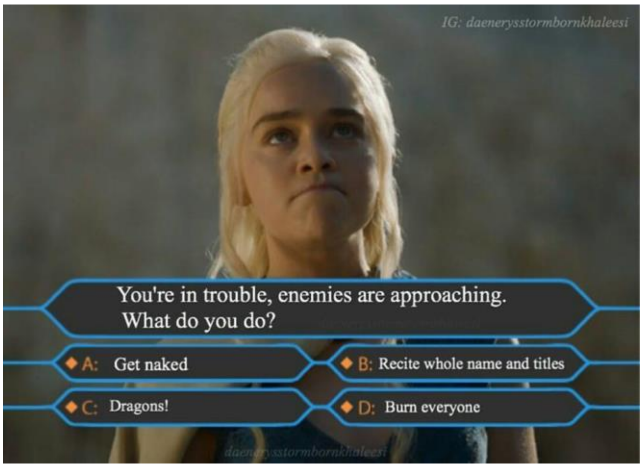

IG: daenerysstormbornkhaleesi
You're in trouble, enemies are approaching.
What do you do?
A: Get naked
B: Recite whole name and titles
C: Dragons!
D: Burn everyone
daenerysstormbornkhaleesi

Figure 1.1 Image based on the TV Shows "Game of Thrones" and "Who Wants to be a Millionaire"

The good thing about this kind of entertainment is that usually, we do not need to know the show to enjoy what was posted and that broadens the target audience. This type of entertainment is only one of the possibilities of the use that can be given to a mobile device. Today, we can already watch television on our mobile devices anywhere we want. We are not confined to our living room like we used to be. This phenomenon is called mobile television, since it combines what traditional television has to offer with the mobility provided by mobile devices [70]. A clear example of the usage of mobile television, is the time spent on public transportation by the youth [17, 53, 55]. During their daily commute, their best source of entertainment is the mobile device that they carry with them.

The mobile television allows users to watch what they want while they are in a public space or in a transportation vehicle. One of the main advantages is when we want to see a different show than others or we need to go to a room without a television set, we can bring our mobile device with us [76]. Usually mobile television users prefer to watch light entertainment to relax after a long day at work (or school) [17]. For this type of occasions, they tend to prefer watching short shows or videos, just to help them spend the time while they are waiting to reach their destination. One thing that people like to do is to interact with content, and like it was said before, that is one of the main reasons for mobile devices being such a great threat to traditional television. But with mobile television, we can now aim to achieve greater goals, like creating ways of interacting with its content (see quiz games [9] for example). We want to explore a new way to interact with shows and the best way to innovate is to use new technologies.

Augmented reality (AR) is a technology that is relatively new in this context, but it is quickly gaining traction and is predicted to be largely used in the future [77]. Some applications can already be found, such as "We Are TV", that combines TV shows with Augmented Reality [99]. This application allows users to interact with augmented objects related with the show but not with the show itself. Our goal is to create a new concept of interactivity with television shows using the concepts of AR and apply it to a mobile device environment. This concept needs to be practical and needs to meet the user's desires. While targeting people from all ages, it has as main objective to attract younger users, since they are the ones that are more likely to use a mobile device for entertainment [100].

1.2. Main Contributions

The main expected contributions of this thesis are:

- Creation and study of a new form of entertainment to enhance the viewer's experience while watching a television show on a mobile device;
- Development of a prototype that will allow users to interact with a television show by using AR to provide them additional information regarding the show or the current scene, as well as to create content based on their interactions with the show.
- Evaluation study to ascertain the users' feedback while using the aforementioned prototype.

1.3. Document Organization

This document is divided into 5 chapters. The second chapter represents the work that has already been done in the area of augmented reality and work linked with television. In chapter 3, we can find the concept that was developed with the execution of this thesis, the various stages of development of the prototype created to demonstrate our concept and an evaluation study for each stage, performed after testing the prototypes with several users. Chapter 4 presents the future work that can be done taking into account what has been studied in this thesis. Chapter 5 contains the bibliography with all the references used to support this document.

2

Background and Related work

This chapter starts by approaching the Augmented Reality area, one of the important parts of the concept that we want to develop. Within this theme, some facial and object recognition techniques are addressed (sub-section 2.1.1) and several research studies developed in this area are presented (sub-section 2.1.2). Finally, in section 2.2 we delve into the television industry and present a state of the art of interactive television, by exploring important themes within the context of this thesis, such as OTT (Over-The-Top) services and live streams (sub-section 2.2.1).

2.1. Augmented Reality

The concept of Augmented Reality was introduced by Ivan Sutherland in the 1960's when he talked about the interaction of a user with computer-generated content [69, 85]. As Milgram's stated, AR is a concept that is related with Virtual Reality (VR) and he classifies these environments based on their content of real and virtual elements (Figure 2.1) [46, 57].

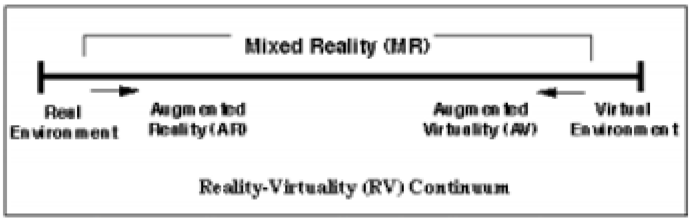


Figure 2.1 Milgram's Reality-Virtuality Continuum [57]

Augmented Reality is a variation of Virtual Reality but, while virtual reality aims to create a new non-real environment, where the users could be immersed and interact with it, in augmented reality the user interacts with the real world and some additional objects superimposed upon it to complement the real world [35, 50]. These complements of reality can be done by creating objects that would otherwise not exist.

When we augment information, we are improving user's perception and improving his interaction with the real world [49]. This interaction, which can be done with augmented objects, gives the user a different view of the world around him, which can lead him to acquire new knowledge thanks to this experience, so that he can perform certain tasks. Some of these tasks may be, for example, to view additional information about an object [69]. We can see an example of this in Figure 2.2.

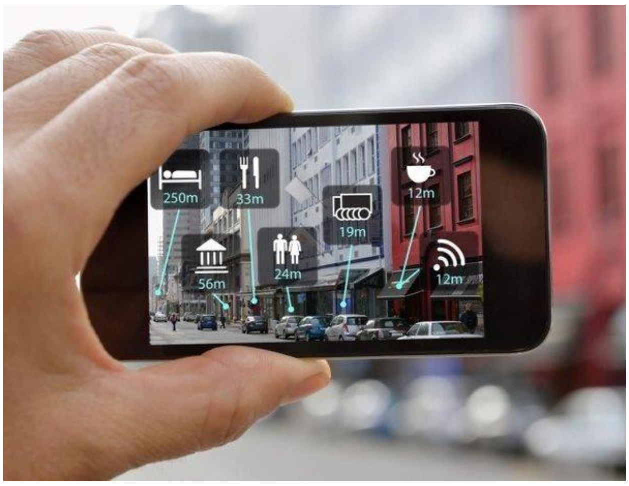


Figure 2.2 Example of AR. [27]

These graphic elements need to be correctly aligned with the real objects that they refer to. Since the graphics presented in an augmented reality system are relatively simple (for example, text, images or an object), it is not necessary to create a 3-D model of all the objects that we can observe. On the other hand, it is easier for the user to see misalignments, since he is experiencing real and virtual content simultaneously [49]. In Figure 2.3, we can see a visual example of the differences between VR and AR.

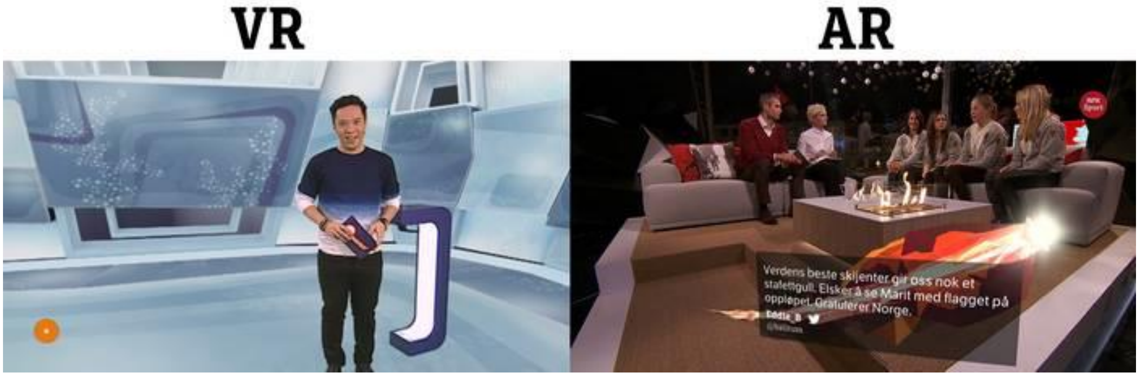

VR
AR
Verdens beste skijenter gir oss nok et stafettgull. Elsker å se Marit med flagget på oppløpet. Gratulerer Norge.

Figure 2.3 VR vs AR [97]

In the past, using augmented reality technology on the mobile devices that we use today, was just a dream. The use of this concept was limited to a single physical location, due to hardware limitations [69]. To solve this problem, Sutherland developed a head-mounted display with the aim to give the user freedom to explore the environment, giving the sensation that the objects are really there, even if he looked to a place that was not initially in his field of view [86]. Thanks to advances in technology for smartphones (such as the integration of high-performance processors, accelerometers, gyroscopes and high-resolution cameras), it is now possible to deploy this technology on a mobile device. Implementing this technology in a mobile phone, gives us the possibility to explore the mobility that our smartphone provides us and create new kinds of mobile applications [69]. The smartphone camera, being small and not causing any obstruction to the user, is ideal to allow us to see a computer-modified version of the real world [67]. The same can be said for other devices such as a tablet since they are similar devices.

However, one of the greatest difficulties of AR is to discover the position and orientation of a real object so that we can perfectly align the graphical overlay generated by the computer with the corresponding objects of the real world [67]. According to the authors in [110], there are three types of tracking techniques: Sensor-Based Tracking techniques, Video-Based Tracking Techniques and Hybrid Tracking Techniques. Although there may be other ways of classifying the techniques (such as [12]), the one mentioned before has gained a wide acceptance by the research community. These are described next.

Sensor-Based Tracking Techniques - These techniques are based on sensors, such as accelerometer or GPS. Each sensor has its advantages and disadvantages [110]. For example, GPS does not work indoors but has an accuracy of less than 3 meters which can help knowing the position of the user [12]. In [25], a prototype that works with GPS and other sensors was created to seek information through a campus, when looking at specific buildings. They also reported some problems with the GPS tracking in some parts of the campus due to obstructions. Inertial sensors, for example, can be used to calculate the velocity and relative position of an object. Some of the advantages of these sensors are that it is not needed to be in line of sight and they work at any range of the tracked object. One of the disadvantages is that the sensors can make the values in the position and orientation drift [12, 26].

Video-Based Tracking Techniques - Computes the position of the camera relative to the real objects by means of image processing techniques in order to create an object in a specific location according to the viewpoint [110]. Over the years, some approaches have been proposed that according to [1] can be divided into two categories:

- **Marker based** - To figure out a camera position it is necessary to have some 2-D points and match them with their 3-D coordinates [1, 110]. These correspondences are created after the detection of the object in question. One of the ways to solve this problem is to detect fiduciary markers. When the marker is recognized, we can proceed to calculate the camera's orientation and position, and an overlay with the 3D representation will occur (Figure 2.4). Even if the camera moves, the image will move accordingly, thanks to this method [51]. Normally, fiducials have some code, and it helps in the process of distinguishing one object from the other [1].

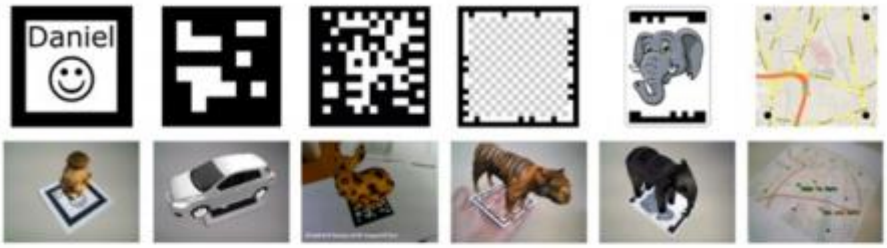


Figure 2.4 Examples of fiducials [72]

- **Marker-less based** - Performing the tracking without markers is a great challenge, since it is necessary to calculate the position of the camera and its orientation through the image processing, that has the ability to detect natural features in a video stream (Figure 2.5). There may be other approaches based on this concept, such as natural feature markers [12], but the model-based technique seems to be the best. This technique consists in identifying features in the images using an object model. Then it uses registration techniques that align the 2D image data with a 3D model. One of the most commonly used features for identifying an object in image sequences are the edges [1].

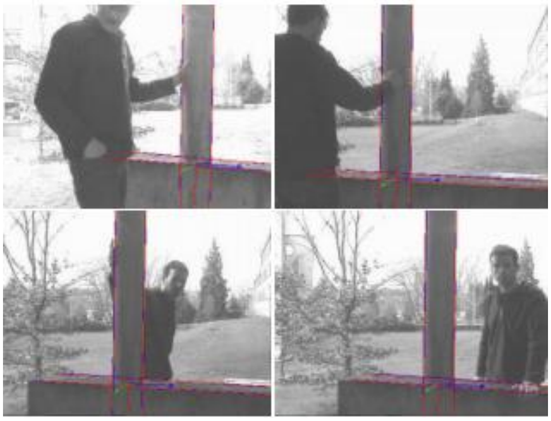


Figure 2.5 Model-based example [19]

Hybrid Tracking Techniques - There are augmented reality applications where using computer vision alone is not enough to solve the problem. In these cases, we have to opt for an approach that combines the two techniques mentioned above. One example is a project [10] that aimed to combine GPS and other sensors with vision for a more robust solution in an outdoor environment [110].

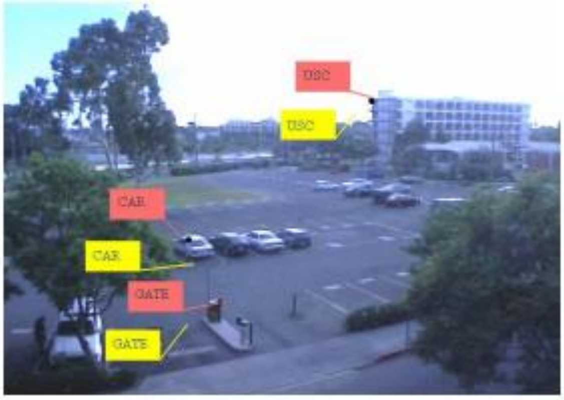


Figure 2.6 Example Hybrid Tracking [104]

As we can see in Figure 2.6, this is an implementation of a hybrid tracking approach. The authors explain that the black dots identify the feature points that they want to track and annotate, the yellow rectangles are the annotations made with the information gathered from their sensor approach and the red ones are the ones with vision-corrected positions.

In our solution, we do not just want to track the objects, we also need to recognize them in order to provide information about them, using AR concepts. This is also applied to faces, which is detailed next.

2.1.1 Facial and Object Recognition

A face can be considered an object, being a non-rigid object. This means that we can assume that the facial recognition problem is a sub-problem of object recognition. With this in mind, it will first be approached the recognition of rigid objects and then we will proceed to the subject of facial recognition [39, 52].

The use of markers (fiducials) turns the task of recognizing any kind of object relatively easy, thanks to its simple use and how robust this method is. They already have information within them about the object that will be augmented to facilitate the job of recognizing an object. The problem is that we need to set-up the real environment [31] to use this approach and that is not viable in some occasions. That said, there are some approaches that can be used to recognize a rigid object without the use of markers, such as using natural keypoints or using natural edges [94].

Using Natural Keypoints

A keypoint is a point of interest in an image. Once a keypoint is selected, this keypoint will be represented as a vector (as we can see in Figure 2.7) [47]. This approach consists in the extraction, description and matching of keypoints.

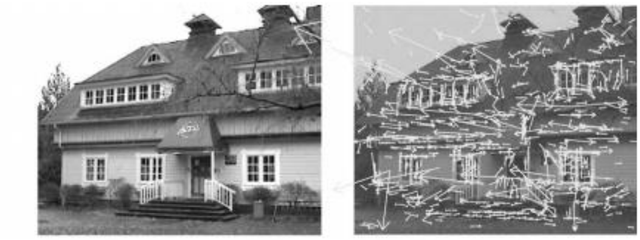


Figure 2.7 Example of keypoint selection [47]

The extraction phase consists in the detection of pixels that are different from other pixels.

In the description phase, a vector with information about the characteristics of a keypoint will be created, so that it can be used in a comparison between two keypoints. There are two types of approaches for local texture-based description, which are the use of gradient histograms or binary tests.

Finally, in the matching phase, it is necessary to perform a comparison with a set of images. In the case of living beings, to identify and recognize an object, it is necessary to know what we are looking for, so that we can create an association and conclude that it is indeed something similar. With object recognition, the principle is the same. To recognize an object, this object must be compared with an existing object model [65]. That said, feature vectors of keypoints related to a reference or model, needs to be stored in a database so we can use it to make comparisons. With this, we can search our database and find the feature vector that has more similarities with that particular keypoint. When we have a large feature vector, the search cannot be done in real time [94]. The solution is searching for the neighbors using a tree-based search [8] or using hashmaps [20].

Using Edges

In this approach, edges are used to calculate geometric features. A method with high level of accuracy was presented by [31] that is able to recognize planar shapes (Figure 2.8). In this method, the authors use the contour concavities to recognize a shape and extract it, being able to calculate the camera's pose.

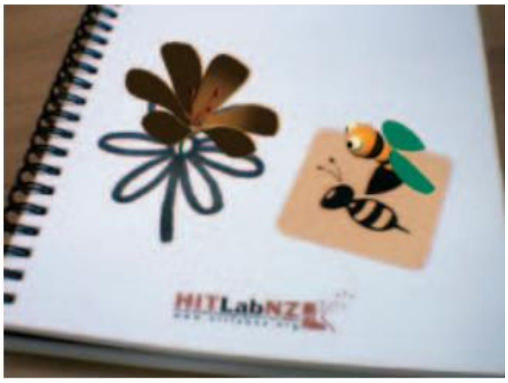


Figure 2.8 Shape Recognition [31]

Facial Detection

The first step in face recognition, is face detection (we can observe the process in Figure 2.9). To detect a face, we need to overcome some obstacles such as pose, presence or absence of structural components, the facial expression, occlusion, image orientation and imaging conditions [103]. Detecting a face on video is not the same as detecting in a single image. In a video, a face will have a lot of positions and will be looking to different angles [98]. To detect it on video, a usual procedure is to detect the face in one of the frames and then track it through the others [98, 111]. In most cases, what happens is that the system is waiting for the perfect frame and when it finds it, a still-image-based recognition technique is used [111]. A good example of this case is [45] where the authors used a template matching method to detect faces. After detecting a face, the authors used their solution to track the detected faces in their neighboring frames, no matter in what position and orientation, through adapted models.

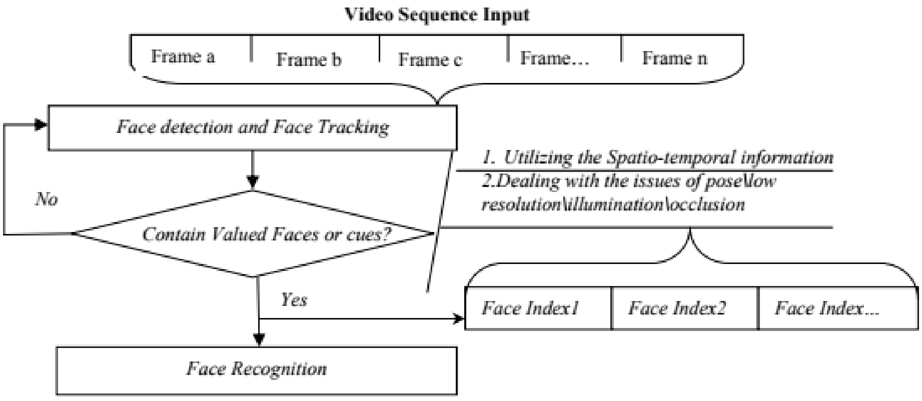

Video Sequence Input
Frame a
Frame b
Frame c
Frame…
Frame n
Face detection and Face Tracking
No
Contain Valued Faces or cues?
Yes
Face Recognition
1. Utilizing the Spatio-temporal information
2.Dealing with the issues of pose\low resolution\illumination\occlusion
Face Index1
Face Index2
Face Index…

Figure 2.9 Process of face recognition on video [98]

After a face is detected, the face will be tracked along the other frames. It is a safe bet to search the face in the same region in the neighbor frames, since it is not very common for a face to be on a very different location based on the previous frame [45]. There are some ways to do this. For example, in [18] non-rigid objects are tracked based on features, such as color or texture. In [36], the authors used a color model and a deformable template. The goal is, based on pixels, find out if there is a face on the image and then the deformable model is used to group the ones that most likely belong to a face.

There are numerous techniques for detecting a face in an image and [13, 103] separated them into four categories:

Knowledge-based methods: This type of method is often used to discover the location of a face. This type of method knows what rules exist on a human face and uses that knowledge to detect a face [13, 103]. A major problem with this approach is to define the optimal number of rules. If we have rules not strict enough, we will have false positives. There may also be the opposite case where due to the strictness of the rules, a face is not identified [103].

Feature Invariant methods: The goal of these methods is to identify features that are present in the structure of a face even if the image has problems with lighting or viewpoint. These methods, like the knowledge-based ones, are used to search for face location [13, 103]. One option to extract these features is to use edge detectors [103]. One major drawback of the edge detectors approach is that the features needed to use it can be impossible to extract in some situations such as shadows creating false edges, providing in this way, wrong information [103]. Some applications of this method use face texture or even the skin color. Studies were made [96] and using the skin color is very effective not only in detecting faces but also in the detection of some body part.

Template Matching Methods: In these methods, templates with patterns of a face are manually defined by a function. Then, these templates are compared with the image we want [13, 103]. This type of method is easy to implement, but it has a great disadvantage. It cannot adapt to the image, so it can often fail due to some variations that may occur. Because of this problem, solutions have been presented to overcome this disadvantage by implementing other types of templates such as deformable templates [103].

Appearance-based methods: In this case, the software learns what is a face resorting to machine learning and statistical analysis. This type of method is used usually for face detection [13, 103]. The goal is that the algorithm discovers what is a face. According to [107], appearance-based methods are better than the alternatives thanks to the advances of modern technology. Some examples of the use of these methods are neural networks [108]. Neural networks are very efficient and there are a lot of solutions with them. Other great example of this type of method is the use of eigenvectors [92].

Facial Recognition

When a face is detected, it will be normalized, to avoid some issues during the facial recognition procedure. There are different ways to recognize faces. Facial recognition is an area that has attracted many researchers so far due to its challenges, since it is not an easy task. However, still image facial recognition can be classified into three categories [109]:

Holistic matching methods - This type of methods try to recognize a face based on the whole image and can be divided in statistical approaches or artificial intelligence approaches [39]. One of the most used approaches, in this type of methods, is eigenfaces [109].

Feature based matching methods - These methods identify features present in the face, fiducial points to some, and compares the geometrical relationship between each point with the information that is stored in the database [14]. In [52] it was designed a system where the features are extracted from the data without assumptions on the appearance of a face.

Hybrid methods - This type of method combines both methods stated above. Some researchers say that by using the advantages of each method, the facial recognition process becomes more reliable [109].

2.1.2 Applications of Augmented Reality

Augmented Reality can be used in multiple situations, such as medical purposes or in the construction area [71], but for the sake of this thesis, we will focus on the entertainment and information area.

There are already some applications in the entertainment area that use Augmented Reality. One of them is the game "Pokemon Go" developed by Niantic. The objective of this game consists in a user walking through a street looking for "Pokémons" (virtual animals) and when they appear in a specific location, the user needs to use a "PokéBall" to catch them (Figure 2.10). A "PokéBall" is a special item that exists in-game and is needed to catch a "Pokémon". In this game, it is also possible to earn "Pokéballs" or fight with others in specific locations. These locations, on contrary to "Pokémons" are in fixed points [23].

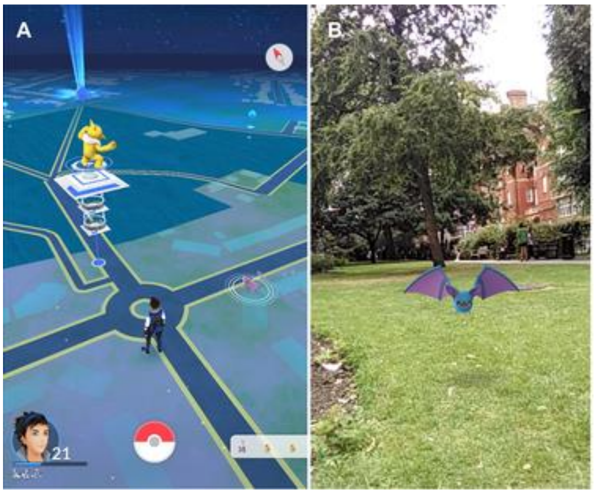


Figure 2.10 Pokémon Go [23]

Some researchers are already studying ways to create games where it is not needed to touch in the device in order to interact with the game. We can see in Figure 2.11, an example of a Football game where the user uses the movements of his foot to kick the ball [50].

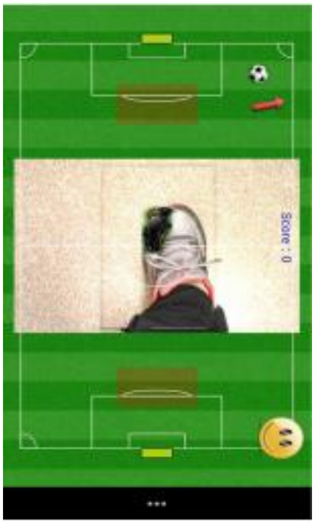


Figure 2.11 Football game using AR [50]

There are already some applications for AR that consists in providing information to the user. Some of these applications aim to provide information about specific places that the user is visiting by putting virtual labels in real objects. These objects can be recognized, or the labels can be put through GPS information and other sensors [71]. One of these cases is "Yelp" (Figure 2.12).

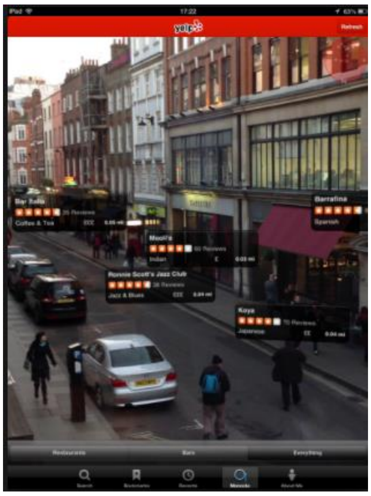


Figure 2.12 Yelp [105]

AR can also be used in a television context, as for example, in a football game broadcast (Figure 2.13). This kind of augmented reality creates some content, that will be put on top of the broadcast, so the viewers can be provided with extra information. This specific example does not give the users the possibility to interact and choose a viewpoint since the game is being captured from tracked cameras [71].

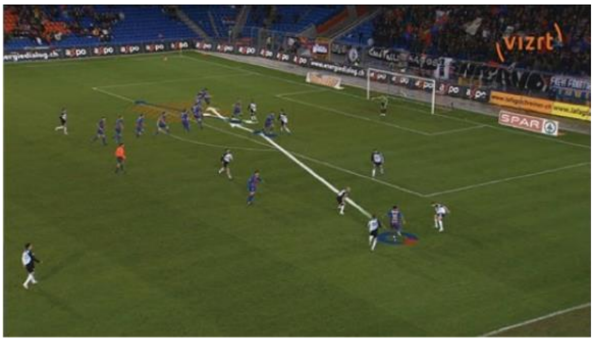


Figure 2.13 Football broadcast [71]

There are already some studies in the area of mixing augmented reality with television broadcast, where the viewer can interact with the show he is watching. In [42], the authors give an example of one of the possibilities of using this kind of systems, where a student can interact with the television to learn more about a specific organ (Figure 2.14).

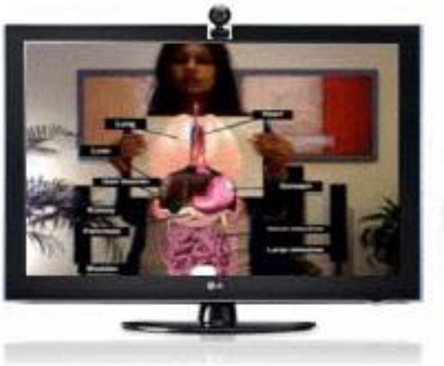


Figure 2.14 AR mixed with Television [42]

Table 2.1, taking into account some of the applications that can already be found in this area, each one with a different purpose, presents a quick comparison between all of them and what are the goals of each one and individual features. After that, a comparison is also made with our concept, so it can be made clearer what are the differences from what can already be found and what are we expecting to bring to the world with this thesis.

Table 2.1 Comparison with other apps

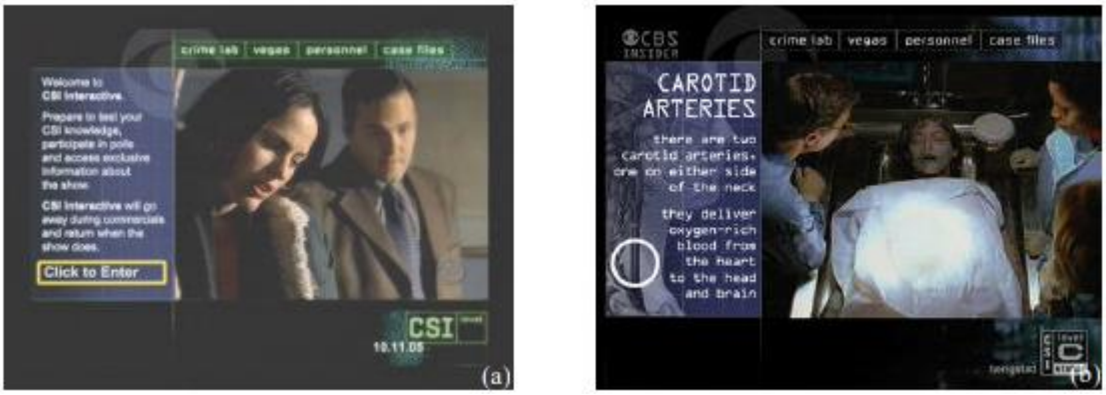
	Only requires one device	Complements content from a television environment	You can interact with the app	Gives the possibility to create content with the modified reality
Pokemon Go	Yes	No	Yes	No
Football Game	Yes	No	Yes	No
Yelp	Yes	No	Yes	No
Football Broadcast	Yes	Yes	No	No
Student App	No	Yes	Yes	No
Our Concept	Yes	Yes	Yes	Yes

2.2. Television

Television, defined by [16], is a device that has a screen and can transmit images and sound through the transformation of electrical signals that the device receives. The use of television is quite controversial, some studies [2, 101] were made through the years to link television with some problems that may affect society. In the past, there were several opinions about this technology. Some said that television shapes our thoughts while others defended that it was something that would not affect the important decisions that have to be made in society. One thing everyone agreed with is that television was used with frequency and sometimes even served as a "background noise" while the audience were doing other activities [37]. This kind of studies did not stop here, and some more recent research affirms that television is indeed contributing to shape the mind of the youth. This can be good or bad, depending on how television is being used [73]. There are also studies that claim that what happens in the news that are broadcasted, have a great influence in what society will regard as the more problematic issue in that moment [38]. All this indicates that in fact, television influences and has a significant impact on our society.

Through this device, one can watch the news or some form of entertainment. Although television has both goals, television viewers usually prefer the entertainment area [90]. Television is already rooted in our daily tasks, it is part of our routine. Watching television is the task that occupies the most time of our day soon after sleeping and work responsibilities [30]. More and more users want to have an experience that is not just sitting quietly in their couch and watch content. They are getting used to have a more active role and to process more information at once than before, since it is becoming common practice to use two devices simultaneously. This creates the need to have more control over what they are seeing [48]. The branch of entertainment is a very competitive one and consequently, there are already many studies to innovate in this area, especially about interactive television [11]. In [11], the authors created categories of television content based on the interaction that viewers have with a television show. There is some research work regarding the use of interactive television in the news branch too, such as [63]. The goal of this research is to give the possibility to the user to choose what news he wants to watch and in what order, giving the user the opportunity to interact with the show.

Interactive television is the name given to any video program that provides some form of interaction between the user and the program he or she is watching [48]. Two good examples are the visualization of additional information while watching a show or the chance to answer to questions asked during the show. Some examples of the use of interactive television are the genres of sports, entertainment or dramas. Sports is said to be the best area to use the concepts of interactive television, we can choose the view that you want thanks to the existence of various cameras filming a match, play trivia games or check statistics. The genre of entertainment, being most evident in quiz shows, takes into account the competitive nature of the show, and creates a favorable environment for the implementation of interactivity. A clear example of this is the possibility for a viewer to become a participant in a quiz show through an application synchronized with the show and answer the same questions presented at the show itself while he is in the comfort of his home. This genre is the one that captivates more people and is the one with a larger fan community, that most probably, will follow all episodes. An example of an application used in this genre is C.S.I Interactive (Figure 2.15), where during the C.S.I show, explanations were provided for what happens in the series and other additional information [48].



Welcome to CSI Interactive.

Prepare to test your CSI knowledge, participate in polls and access exclusive information about the show.

CSI Interactive will go away during commercials and return when the show does.

CAROTID ARTERIES

there are two carotid arteries, one on either side of the neck

they deliver oxygen-rich blood from the heart to the head and brain

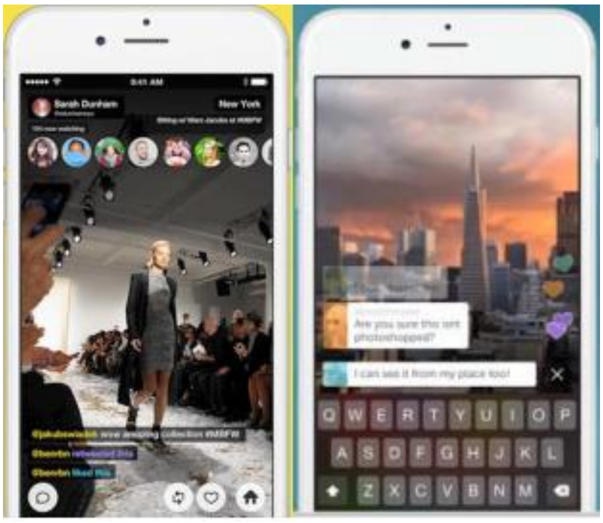
crime lab | vegas | personnel | case files
Click to Enter
CSI
10.11.05
(a)
CBS INSIDER
crime lab | vegas | personnel | case files
CSI
C
(b)

Figure 2.15 C.S.I Interactive

Internet is having a major role on the evolution of television. Now, video content is being broadcasted over Internet. This kind of delivery can be called as an Over-the-Top (OTT) service. An OTT service is when media content is provided through the Internet's infrastructure that already exists but is not controlled or provided by a telecommunication provider. This type of service is a threat to the existing telecommunication services, since they can provide the same content to the user [43]. Netflix uses this service to stream its video content [82]. This market is on the rise [80] and with the existence of these services, exists the possibility that people end their television subscription and start using solely this kind of services, since they can find the content they want to watch online [82]. Due to these recent events, some television networks are also adapting and creating applications to be able to stream content using OTT services. This is the case of HBO and CBS that already have services of this kind [22, 95].

2.2.1 Live Stream

Live streaming can be defined as a video being broadcasted over the Internet in the moment that it is being made [15]. This kind of broadcast" combines high-fidelity computer graphics and video with low-fidelity text-based communication channels to create a unique social medium" [33]. Live streams are a popular mean to share information about some breaking news or special events [32, 87]. The great difference between live streaming and a video is that in some streams, the viewer interacting with the streamer or other viewers, can alter the course of what is happening in the broadcast. It is possible for a viewer, to interact with the stream through a chat or any other ways that can be implemented (Figure 2.16). In [32], this interaction is described as necessary when watching an event so that a user can take advantage of the whole experience in order to simulate what would happen if the person was present. The content presented in a live stream is very diverse, can range from the streamer talking with the viewers to a stream where something is shown or taught [88].



Are you sure this isnt photoshopped?

I can see it from my place too!

Figure 2.16 Live Stream example [88]

A downside of a live stream is that when there are too many viewers interacting, the chat may experience some problems and it can even lead to persons not reading all messages. Some research works are already being done to fix this problem [58]. There is also some research work that aim to mix AR with live streaming, such as [61]. In [61], a study was made to create a way of showing comments in the live video resorting to Augmented Reality. Another interesting prototype that was developed in this area, was an application that not only allows to interact with the stream, but also gives users access to a specific menu where from a preview of some live streams, he can choose in which one he wants to participate (Figure 2.17) [89].

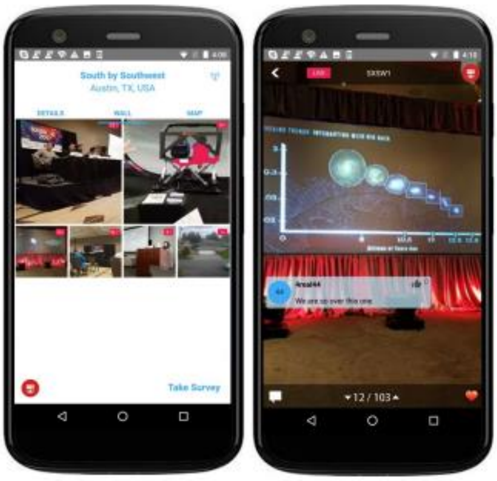


Figure 2.17 Android Prototype [89]

One of the most popular sites that provides live streaming is Twitch.tv. In this site, usually a person streams himself playing a specific video game for others to watch and comment. It is even possible to show the game and simultaneous show the face of the streamer while he plays (Figure 2.18) [33]. This interaction helps to create a new type of social community, that is regarded as very active in a lot of social web platforms [41].

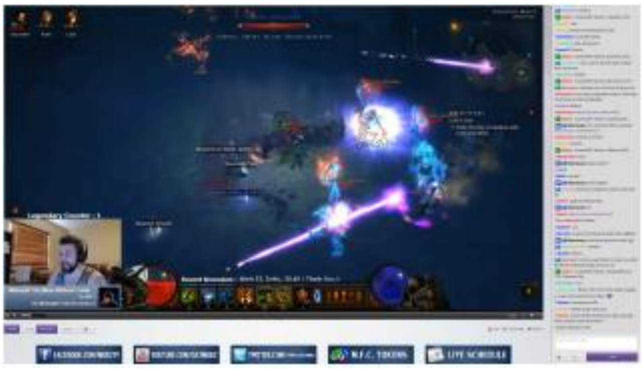


Figure 2.18 Twitch [33]

Although this kind of entertainment is not so active as playing the video game itself, it is way more interactive than traditional television, since the viewer can interact with the person that is broadcasting the live content [74]. The channels are not always online, the streamers are not always active, and the channels depend on their upload. This could be considered as a problem, but on a service like this, there are always active streamers ensuring that content always exists to be consumed. There are other kinds of service where the target audience is not intrinsically the same as Twitch, like for example, YouTube Live [64]. These types of entertainment are shaping the world of entertainment as we know it, creating even new types of jobs, such as full-time streamers. These persons can have some remuneration based on some factors, such as having some kind of advertisement in their channels or the number of viewers that subscribe to them [33]. Some researchers are already studying these communities, to know how they can apply these concepts to the interactive television community [75]. The concept that will be developed can be applied in a live stream environment. For example, a user creating content can at the same time be streaming his actions so that other users can watch his creative process in real time. Viewers can also interact with the streamer and give feedback regarding his actions.

3

TvTeller: Your way

As already mentioned, the experience of simply watching television is out of date. To solve this problem, we developed a concept that aims to innovate this whole experience giving more interaction power to the user. Due to the fact that, increasingly, the use of mobile devices exceeds the use of television among young people [81] and since the number of persons that use these same devices to watch videos is also increasing [79], the opportunity to create a new kind of experiences arises.

Television has the ability to entertain a viewer or simply to provide unidirectional information. Unless the viewer switches the channel or the TV show, he cannot choose what to watch. As for the entertainment part, the entertainment is very passive. The maximum that the user can do is choose which show to watch. There is nothing that requires the user to pay full attention to what he is watching. This gives room to the user to perform other things simultaneously, which may divert his attention from what he is watching. We aim to change that by giving people the chance to interact with what they are watching. Using a mobile device as a main screen, viewers will have the option to choose what information they want to see about what they are watching, or if they want to see it at all. Then we want to combine the interaction that is possible to have with a mobile device with the show that is being watched, giving the user the opportunity to create his own entertainment product instead of only watching the creations of others.

Given the impact that social networks have in our daily lives, where people like to share funny images and videos between each other, and the possibility of watching television in a mobile device, we decided to develop TvTeller. The goal of TvTeller is to provide the user a whole new experience, complementing the "passive" TV experience. We aim to give the user the possibility to know more about his favorite characters, create amusing content based on the show, and share it with his friends.

3.1. Concept

TvTeller is a system that can complement the experience that the viewer has when watching a certain series, making it more interactive and social. This concept is composed by two components that are independent from each other. One is more information-oriented and the other more entertainment-oriented. Thanks to these two components, this concept innovates the act of watching television giving the user the possibility of doing something more than passively watching television content. To achieve this goal, Augmented Reality concepts were used and combined with the experience of watching a TV show on a mobile device. Instead of adding virtual objects to the video coming from a camera, we will add these objects to the video coming from a local file (although the concept also works with a video stream URL).

The information-oriented component has a more personal aspect. It gives the user the possibility, while watching a series, to know more about the characters or some other important elements, which includes animals and objects that appear on screen while they are watching the episode. This can be done thanks to facial and object recognition where the characters or objects are detected and recognized. The user has the opportunity to trigger an informative virtual object that appears superimposed on the video with the information, so he can learn more about what he wants. This part was thought for the people that like to know more about what they are watching. Currently, if for some reason a user wants to know more about what he is watching, he needs to search it outside the TV context. If he searches while he is watching the episode, he must divert his attention. This is also good for users that tend to do some activity while watching an episode simultaneously. If the user is not understanding something due to not watching some part of the prior episode, he can access the information that is provided to catch up.

The entertainment-oriented component is a more social one. It gives the user the chance to trim part of the episode that he is watching so he can edit it later. This trim has the intention of being non-intrusive. The viewer can trim a moment that he liked while he continues to watch the episode, not ruining his experience. After the trim is done, the user can edit the trimmed video adding virtual objects to the characters that appear in it. The user's creations can be shared later on a social network. This aims to give the viewer the possibility of providing comic relief to him and to all his friends. Since sharing content that is considered funny is something very usual nowadays, in a way, we think that the user will feel motivated to watch the episode with more attention, so he can find the perfect moment to edit.

Although this concept applies to any series, the series "Game of Thrones" was used as the case study. This series, due to its popularity and its unique characteristics, becomes a good example to show our concept. For example, it has a lot of characters, so it is easy to forget who someone is and if he did something that altered the course of the plot. Besides, it is a show with some recasts, which can confuse a viewer when he sees a new actor for the first time doing an existing role. This concept can also be used in movies (especially when the movie has its own universe), but it will have more impact in a series, since the user needs to follow the episodes and is more likely for him to forget about something that has happened in an earlier episode. Although it is not intended to be used that way, the concept can be applied to almost every TV content, such as reality shows, or comedy shows. The information-oriented component can also be used in a documentary context about animals to show more information about the species. Although they do not possess a face, we also detect non-human objects, considering that a face is a non-rigid object.

One possible implementation is the following scenario: the user is watching the “Game of Thrones” TV show and a character appears on screen. The character was recognized, and the viewer wants to know more about him, so he touches him (this is useful when a minor character appears, and the viewer needs to search information to know who he is). A virtual object appears superimposed on the video with the information and the video stops to not ruin the experience while the viewer reads what he wants to know. After that, the video starts again, and everything is back to normal. After a while, the user likes something that happened and wants to edit that moment. The user trims the moment that has just passed, goes to the gallery, and starts editing it, adding glasses to the characters or adding thought balloons to give a different meaning to that specific moment. In the end, the user loves his creation and decides to share it on a social network, so his friends can see it.

With the idealization of this concept we seek to allow users to interact with the content presented on the show, thus overcoming the limitations of traditional television. Given the popularity of mobile devices and their interactivity capabilities, we have chosen to use them as a platform for this new concept. This also gives the possibility for the user to enjoy this kind of entertainment in any place. Since people like to communicate with each other through posts on social networks, we complemented this idea with the possibility to post online the results of such interaction. This way, viewers can share content with others and that will encourage the creation of more content, generating a reliable source of entertainment.

In this research, we evaluated the possible scenarios where this concept can be applied, researched what techniques could be used and evaluated the success of the concept.

To test our solution, we created a mobile application capable of doing what we proposed so we could evaluate the outcome of such interaction and whether the users' feedback was positive. This application was developed in Swift and it is necessary a device running iOS 11.4 (minimum) and a minimum of 1.5 Gb Ram. We chose to develop for iOS because of the resources that are already provided in facial detection area and image classification, such as Vision [7] (Apple's computer vision framework to detect all kind of objects). This provided us with the necessary resources to detect images and create virtual objects to augment and add them to the video. Since tablets' screen are bigger and better for watching videos, the mobile application was developed for tablets, but it should run on smartphones too. This concept was tested in a controlled environment. Before creating the mobile application, first we had to analyze what would be the requirements of an application like this. Then we created a first prototype based on the requirements that we decided that would be necessary and made an evaluation to the prototype resorting to user tests. Only after that, we used the results from that evaluation to solve some issues and created the computational prototype. This prototype was also tested by users, so we could evaluate the success of this concept and solve some potential issues that could not be discovered on the first one due to its limitations.

3.2. First prototype

When we are trying to create a brand-new product or concept, it is usual and a good policy to create some prototypes first. Usually, the first prototypes that are designed are made in paper, since they do not cost anything except the time that is required to make them. While someone is creating a paper prototype, the ideas to implement in the final product can also flow better, since the person is already watching some results and the ideas are not only in his mind. After the creation of these types of prototypes, usually they are submitted to an evaluation, for example, users testing the prototype, so it can be evaluated how easy it will be for someone to perform some task and if the final product will meet the users' needs and preferences.

Since we are working with paper, it is easier to make modifications to the product, which if needed on a later stage it would be much more expensive and difficult than simply making another drawing.

Although these early prototypes are often made in paper, nowadays, they can be done on a computer, using specific applications or websites for these effects or simply by drawing on the computer. With this more modern approach, we can spare paper and it is also easier to change some mistake that in paper would make us start from the beginning. While doing this first prototype on the computer, it also gets easier to understand how the final product will look like. We will have a drawing of something that can be more similar to the specific interface that we are trying to create. Taking all this into account, we decided to make this prototype using an application called Marvel [54].

At first, we defined what should be the application features and what we wanted in concrete. After that, the creation of the first prototype started. Since we were using Marvel to create the prototype, all the interaction that the user could have with this prototype was as close to the real product as possible, at least in this initial phase. We designed the prototype to be intuitive and user friendly. It is a high-fidelity prototype in look, and although this is not a full working prototype, everything on the interface seemed real. This prototype is composed of pages with drawings although it may not seem. When you click on a specific zone, it changes the page to simulate the interaction with a real application.

Taking into account the application features, we tried to make the prototype the most similar as possible to what iPad users could expect. Although the prototype has 33 pages to give the most credibility possible, the application only has 5 views. The rest are the simulations of changes made to one of the main views. These 5 views will be explained below.

The **Home** view, as we can see in Figure 3.1, has two buttons. The user can choose whether he wants to watch an episode from a series or go to the gallery where he can find the clips that he has trimmed earlier so he can edit and share them.

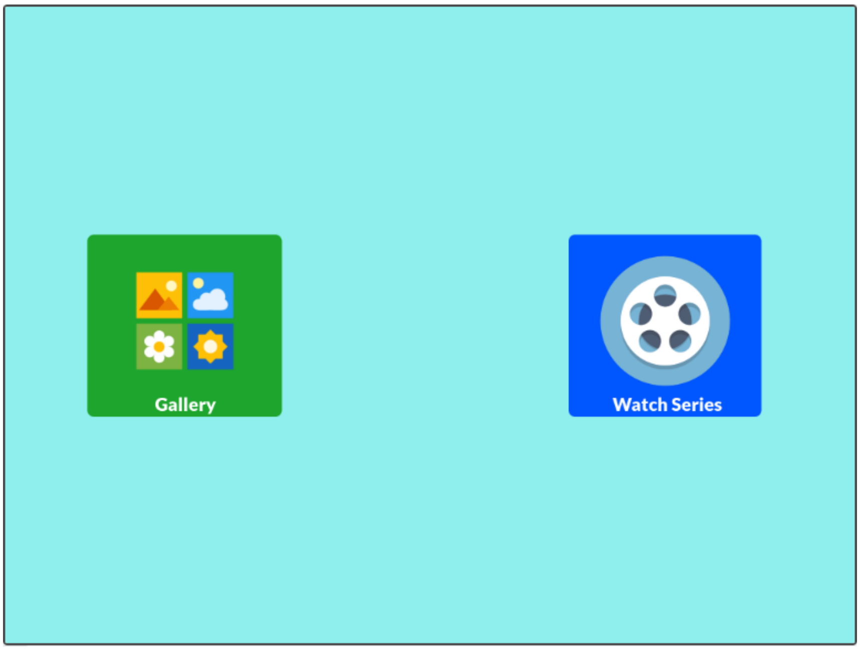


Figure 3.1 First Prototype – Home view

The **Series** view allows the user to choose a series to watch, as we can see in the Figure 3.2. For testing purposes, only 3 series are shown and the only button that will trigger an action is the one from "Game of Thrones". After clicking in the button, the user goes straight to the video player.

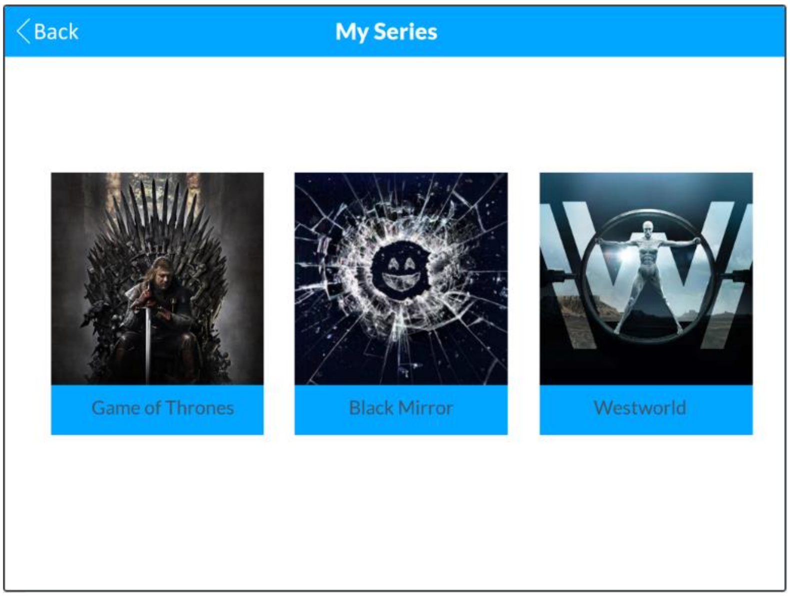
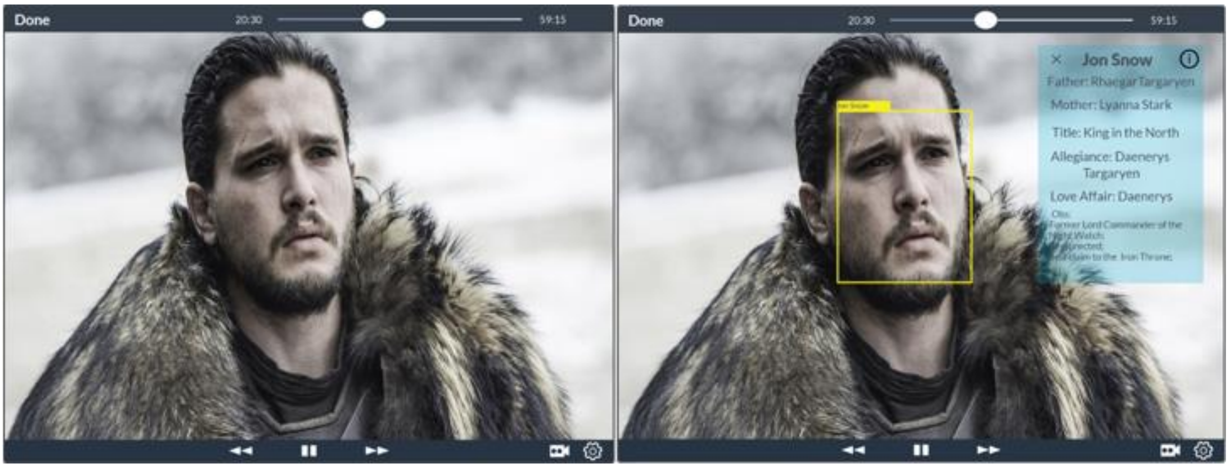


Figure 3.2 First Prototype – Series view

The **Video** view (left side of Figure 3.3) is where the user can watch the video. In this view, the user can also activate the recognition feature, clicking on the wheel gear on the right bottom of the screen. After the user clicks on the wheel gear, a menu appears, and the user needs to click on the switch to activate facial recognition. If this feature is activated, a square will appear over the face of the characters where you can click to show information, as you can see on the right side of the Figure 3.3. We tried to mimic the video experience so when the user is on this page, we have 3 images that are always changing between each other every 5 seconds to fake the action of the video playing. When the user clicks on a face of some character, this stops while he is watching the information. The user can watch information about the character but also about the actor that plays that specific role.



Father: Rhaegar Targaryen

Mother: Lyanna Stark

Title: King in the North

Allegiance: Daenerys Targaryen

Love Affair: Daenerys

Obs:
Former Lord Commander of the Night Watch;
Resurrected;
[illegible] claim to the Iron Throne;

Figure 3.3 First Prototype - Video view

Our concept also gives the possibility to have information about non-human features. To show this, an image of a dragon appears, and the user needs to click on an icon on the top right of the screen to watch the information, since dragons have no face (Figure 3.4). This icon is often used in games to show a player that there is something to interact. At last, in this view the user can also trim the moment that he wants to edit later, using the button that seems like a camera. After the user clicks on this button, a notification will appear, so he knows what happened and what he can do next.

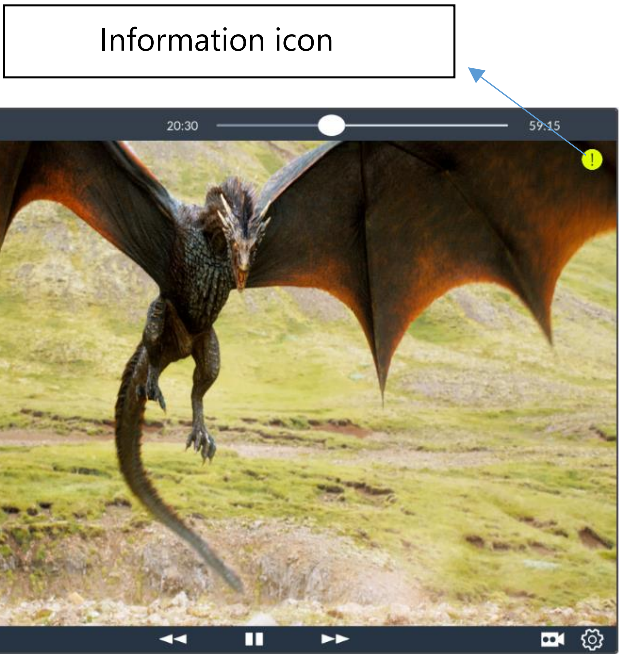


Figure 3.4 First Prototype - Video view 2

The **Gallery** view, in Figure 3.5, is where the user can find all the moments he saved.

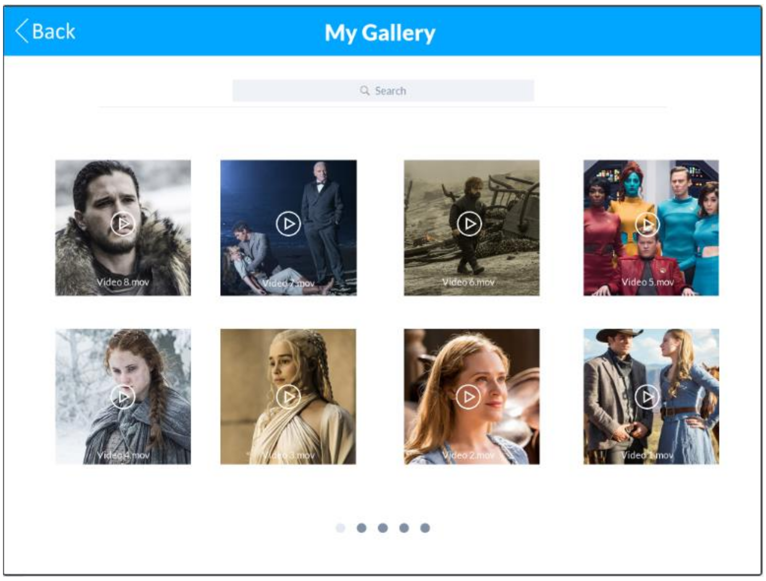


Figure 3.5 First Prototype – Gallery view

At last, we have the **Editor** view, where the user can edit the video, as can be seen in the left side of Figure 3.6. Here, the user can choose from one of the available virtual objects to superimpose on the character and change his appearance. After he clicks on one of them, the image changes to mimic what would happen in a real application. The first button, the thought balloon, allows the user to write something in the balloon (in this case, the text is already written, as it can be seen on the right side of Figure 3.6). After the editing, two buttons appear on the right side of the player where the user can save the new video (left button) or share it on a social network (right button).

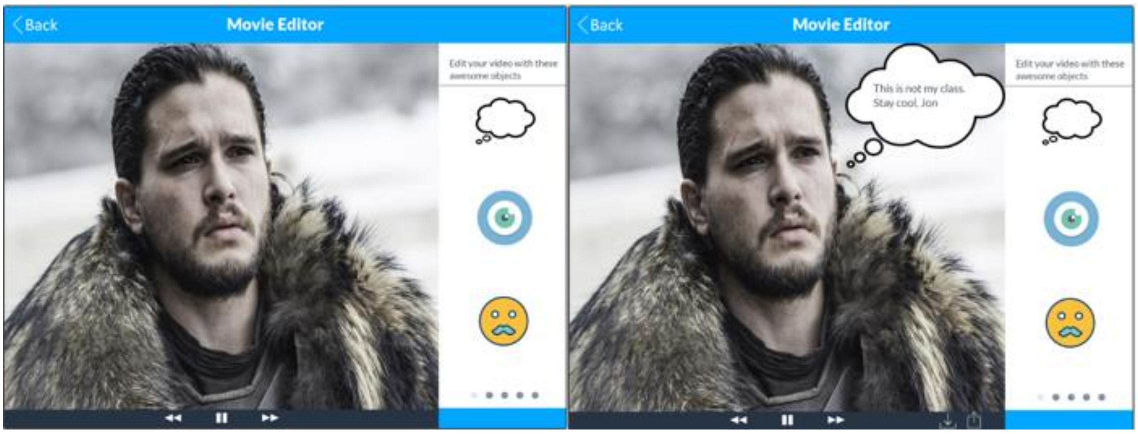

Back
Movie Editor
Edit your video with these awesome objects
Back
Movie Editor
This is not my class. Stay cool, Jon
Edit your video with these awesome objects

Figure 3.6 First Prototype – Editor view

3.2.2. Evaluation

After the creation of the prototype, our main concern was its usability and usefulness. This was the best phase to make big changes if needed, before we go even further, since from now on, all design changes would not be so easy to implement. Taking this into account, we made usability tests to see how target users, that have never seen or heard anything about this concept, reacted and interacted with our first prototype. If some change related with the interface needs to be done, hopefully it will be realized in this phase thanks to observing what users do and what are the mistakes that they make using the prototype. To ensure that the experience is as close as possible to the final product, this prototype was used on an iPad.

Participants and Methodology

The usability tests were performed by 26 participants with ages between 21 and 49 years old ($\bar{x}$= 24,6), where 8 were female and 18 were male. While one participant was performing the test, no other participant was in the room to ensure that his actions are not biased. The tests were performed by a researcher that had the task of observing the users and leading the tests. This researcher had to give a small introduction to the users and had to take notes during all the tests regarding the mistakes that were being done. The researcher never helped the users while doing the tasks. The users were also encouraged to think aloud while taking the tests, talking about why they were making that type of decisions and what they are liking or disliking. Each user was asked to follow some tasks, as you can see in table 3.1. After completing all the tasks, the users had total freedom to explore the prototype and play with all virtual objects. When they finished exploring the prototype, they underwent an informal interview.

Table 3.1 First Prototype - User Tasks

Tasks
1. You enter the app and want to watch an episode from a series.
2. You want to see an episode from Game of Thrones.
3. Since you do not know the characters, you decide to turn on the character recognition, so you can see some details about them. 3.1 While you have the recognition activated, you are curious to know more about the character. You decide to see more information about it. 3.2 You like the actor / actress who plays the role in question and decide to see the information about him/her.
4. Now that you have seen the information that interests you, you want to resume normal viewing and close the additional information.
5. Meanwhile, a dragon appears, and you want to see additional information about it.
6. You already have seen everything you wanted to. Now, you close the information box and return to the normal viewing of the video.
7. While watching your episode, you particularly enjoyed a scene with "Jon Snow." You decide to trim the scene, so it can be edited later.
8. You decide to leave the episode to work on the video clip you just trimmed.
9. You go to the gallery, where you can find all the clips you've ever done before you can edit one of them. 9.1 Then, choose the clip you just saved so you can start editing.
10. Considering the facial expression of the character, you decide to add a thought bubble with a personalized phrase made by you.
11. You loved your work and feel that you need to show it to the world. As such, you decide to share your creation. 11.1 You decide to share it on Facebook

Results and Conclusions

Thanks to the usability tests, we could have a clearer idea on how users will use the computational prototype. It can be said that the results of the performed tests were quite satisfactory. With varying degrees of difficulty, all users were able to complete all the tasks. Most errors were related with the gear wheel where the facial recognition is and the button to trim the video. Participants did not click these buttons right away, they had to think or experiment first, especially to activate the facial recognition.

Some users complained about having to click on the gear wheel button to access the option to turn on the face recognition, claiming it was not intuitive. Some even advised the researcher to replace the gear wheel button by a new button with a more appropriate icon where the user could activate the facial recognition with only one click. A common mistake was that when people saw the dragon, they tended to click on his snout, due to the way they accessed the information about the human characters, instead of pressing the button that appeared on the right corner. Finally, two users also had difficulty distinguishing the share button from the save button on the device.

One thing that was also noticed was that Android users had some difficulty locating iOS styled buttons, such as the button to exit the video player. Although they would eventually click on the right button, they took longer than a regular iOS user, as they are not accustomed to it. At the end of the tests, during the brief conversation with the users, they were encouraged to give suggestions to improve the prototype in its next development phase. There were immense constructive criticisms, in particular regarding the icons' buttons (like the recognition button mentioned before), which were essential in the creation of the computational prototype. When asked about their opinion regarding this concept, all users loved the idea and would use it for sure if it existed. From all the users, 7 reported that when they are using a mobile device to watch a series, they usually need to stop it and search for information related with some character or search after the episode ends so they can understand everything that they have seen. The feedback of the video editing part was also extremely positive with all the users saying that it was great to be able to save moments for later edition. According to the results of the usability tests, almost everyone was amused with both information and entertainment components of this concept and loved the idea and the possibility to have this kind of interaction with TV shows. With these results, it was worth to continue the development of the concept. We proceeded to the next phase taking into account the opinion of the testers, so we could make the necessary improvements.

3.3. Computational prototype

Taking into account the changes that needed to be made from the first prototype evaluation phase, we started planning the development of the computational prototype. This prototype was already fully functional and was possible to interact with it to better understand what the concept behind this thesis was. After the creation of this prototype, we performed usability tests with common users in order to evaluate the viability of the concept and fix small errors that would may appear. This prototype runs on an iPad, where the user has total freedom to interact with the application. All of the previous discussed features were implemented.

3.3.1. User interaction

As noted in the first prototype, this application is divided in two components, the information-oriented and the entertainment-oriented. The application has the same 5 views that had in the previous phase. These 5 views can be divided in 2 sections: the one where the user can watch an episode and the one where he can edit his clip. Both of these sections have their specific features that will be talked about.

After the user enters the application, he will be presented with the **Home** view (Figure 3.7). This view is not very different from the one shown in the previous phase. The major differences are all about the design that was adopted to make the application more appealing to the users.

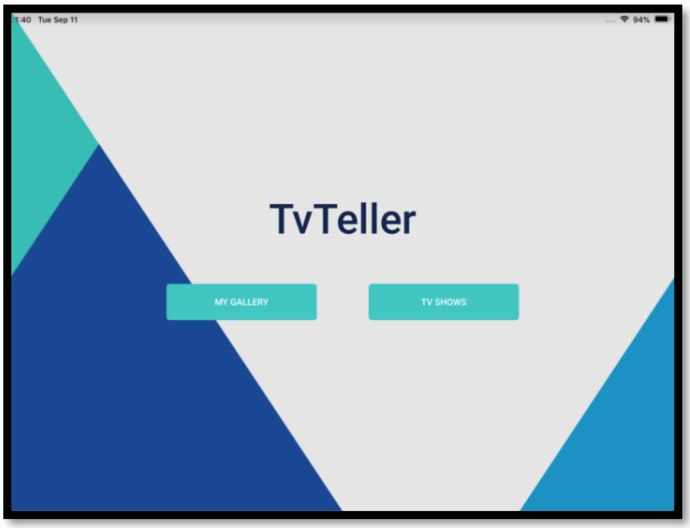


Figure 3.7 Computational Prototype-Home view

On the **Series** view, only minor changes were done (Figure 3.8). Now, the series are in a grid to show how it would be if we had a lot of them, but for the sake of simplicity, in this prototype, we only have the same 3 examples with only one episode working.

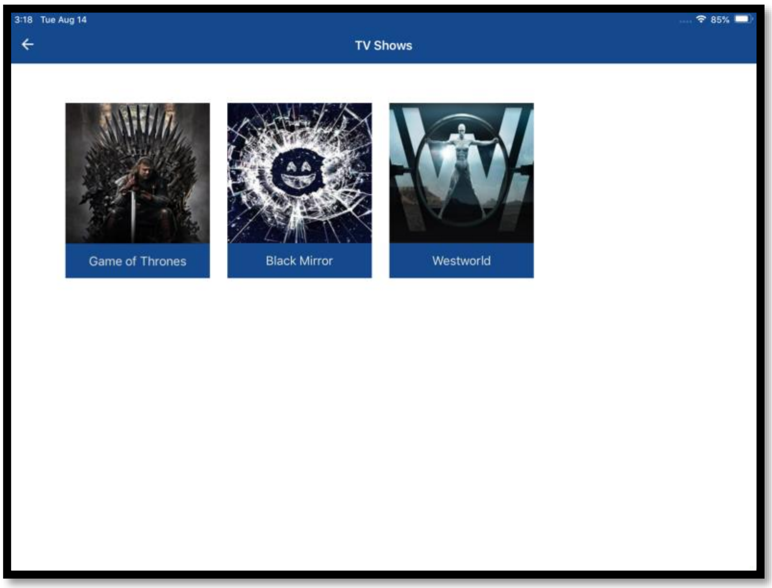


Figure 3.8 Computational Prototype- Series view

In the **Video** view, the number of changes is already considerable. At first, to make the app more intuitive to all users (Android and iOS), and considering the difficulties that Android users had when testing the first prototype, the player layout was made more generic. Now, the time slider and all buttons related to the player are on the bottom of the screen, on what can be called the player bar. The button to leave this view was also replaced by a back button (white arrow), which users are more familiarized with. The buttons that were causing more mistakes, the "Trim Video" and "Facial Recognition", were moved to the navigation bar and the icons were changed (scissors icon and face icon), as it can be seen on the left side of Figure 3.9. The recognition button now is more direct, you do not need to click a general button, such as the wheel gear on the first prototype, and then activate the facial recognition (only one click is necessary). To facilitate, when a user first enters in this view, a tutorial appears to show him the meaning of these two buttons.

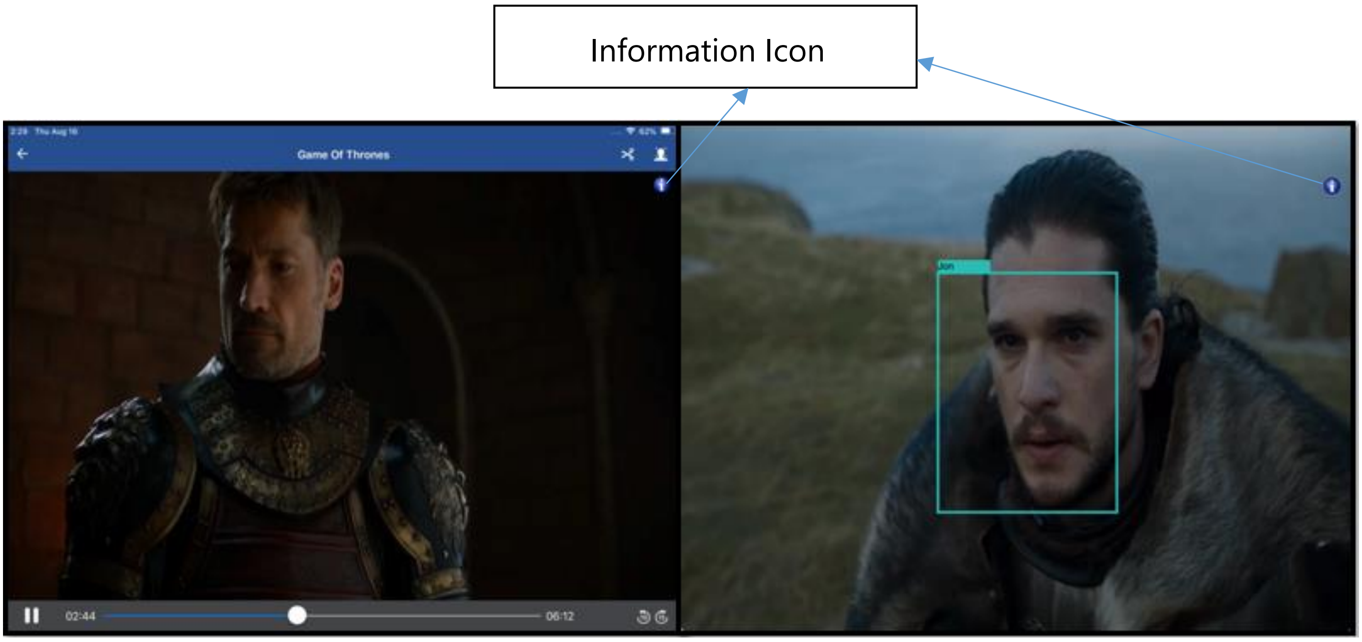


Figure 3.9 Computational Prototype-Video view

The navigation bar and player bar disappear after some time idle to let the user watch the video in full screen. When a user likes a specific moment that has happened, he can trim (save) it. While the video is being trimmed, an algorithm of facial detection is being run to detect the facial landmarks of everyone that appears on the trimmed scene. This can take some time, so a progress bar appears under the navigation bar, to show the user the state of the trimming. When the navigation bar is hidden, a notification will show up at the end of the process telling the user that he can go to the gallery to edit his clip if he wants to. This notification always appears.

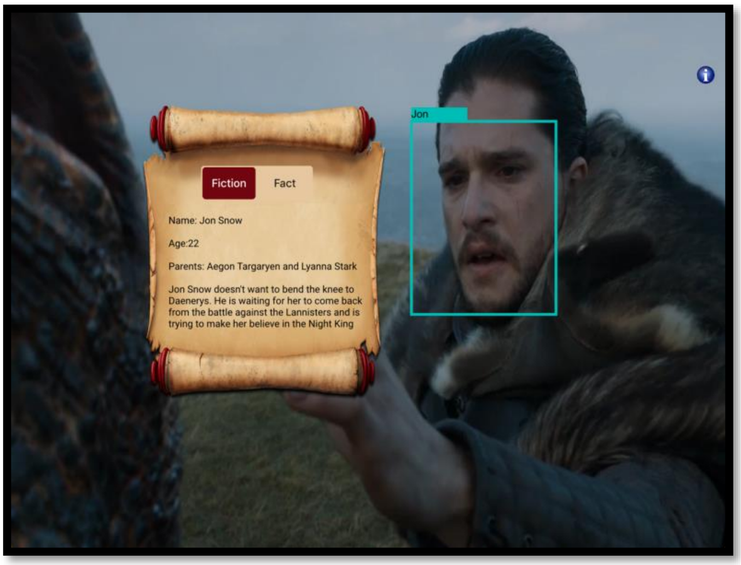
In order to facilitate the user's task (to trim the video), the user only has to click on the trim button when he watches the moment he wants to save. The trim will record 15 seconds of the video: 13 seconds before the button click and 2 seconds after (preventing the desired clip to end abruptly). This duration seems to be enough to catch everything that happened in the moment that the user wants to store.

The user can activate the facial recognition at any moment, and when it happens, a non-intrusive notification appears to let him know that the recognition mode is on. The same happens when this feature is deactivated. If the recognition mode is on when a character first appears in the episode for 3 seconds, and he is recognized with a certain grade of confidence, a recognition box appears over the character's face with his first name (Left side of the Figure 3.9). If for some reason, within this time period, the facial recognition fails, the box disappears, appearing later when the character is recognized. It has been decided that the box appears only in this short time so as not to ruin the video preview. We think that this time is enough for the user to notice that clicking on the face can trigger the appearance of an object with information (Figure 3.10).

The recognition box is not the only way to tell the user that something is being recognized on screen and that the information about the recognized character is available to be shown. A symbol on the top right side of the screen appears whenever a user can interact with a face or a special element, such as an animal or object (Figure 3.9). If this symbol is on the screen, the user knows that he can trigger the appearance of additional information. This symbol was created with the goal of preventing the user from clicking on the screen when nothing is being detected and recognized. It also informs the user when there is a special element on the screen, since these elements do not possess a face, so no recognition box is displayed. This way, the user knows that he can only click on something that is appearing on screen when this symbol is visible, and the recognition box does not need to be always on screen.

We also solved the problem of users clicking the dragon's snout to see information instead of the symbol (during the first prototype evaluation, Figure 3.4). Now, when the information symbol appears, and its appearance is related with a special element being on screen, the user can click anywhere on the screen. If the user clicks on the dragon's snout, for example, the corresponding information will show up. If for some reason the object appears in the same frame as another character, only the character's information will be shown and the information about the special object will appear in other occasion. We made this decision because otherwise the user would think that only the face had information and would not click the element or in case of just the face having information, the user would click on the special element without anything happening. Although this may not be the best way to show information about non-humans, it proved to work as intended since these special elements tend to appear alone on screen.

The UI elements that are used to show information about the characters, are supposed to adapt to each series. In this case, since the series has a medieval theme, the chosen object was a scroll (Figure 3.10). In this scroll, that resembles an augmented object, the user can choose to see the information about the character, with information about what he has been doing lately, or information about the actor. The information about the character is updated by episode so the user can catch up if he lost something important.



Name: Jon Snow

Age:22

Parents: Aegon Targaryen and Lyanna Stark

Jon Snow doesn't want to bend the knee to Daenerys. He is waiting for her to come back from the battle against the Lannisters and is trying to make her believe in the Night King

Figure 3.10 Computational Prototype - Show Information

In the **Gallery** view, the only big change is the way that the trimmed videos are presented. Two of the users did not like much the idea of the videos being numbered, they thought that the order was not clear enough and may cause some doubt. So, instead of being named by number, they are now named by the date they were created (Figure 3.11).

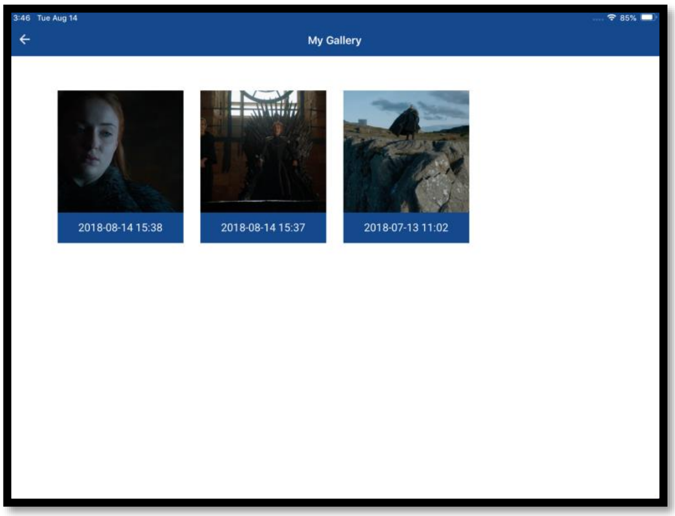


Figure 3.11 Computational Prototype – Gallery view

In the **Editor** view, the design was changed to match the rest of the application. Only three buttons were available in the first prototype, but we wanted to give more options to the user, so he can better experience more what we intend with this part of our concept. The size of the video was also changed to a proper ratio.

As we can see in the Figure 3.12, now the user can choose from 8 different virtual objects that augment the clip. From these 8 objects, the first three (on the left side) are the face objects and are added to the faces of all characters that appear on the video. The other objects are the bubble objects. These objects are added near a face of a specific character chosen by the user.

Due to the confusion that was being made with the "Share" and "Save" button during the tests, we used the normal button in iOS that does both things, since users are more familiar with it. A button to remove all objects that are added to the video was also created (see the bottom of the screen) so the user can remove them if he does not like his creation.

In this view, if the user selects a face button, a moustache or glasses will be added to all characters in the video. If the user selects one of the bubble buttons, an image will appear on the screen (with the shape of the object) and the user will need to drag it to a face. The image will have a message instructing how to be used. After the user drags the image to a face, he clicks on it to bound the object to the face and the creation of the new video will begin. These bubble objects will appear near the face that was selected in that specific time for 1.5 seconds (Figure 3.12). In the thought balloon, the user will have the possibility to write a text. The balloon will also appear 3 seconds instead of 1.5, to give the users time to read the information in it. All these virtual objects are bound to the landmarks of the faces that they affect. After adding an object, the user can remove the objects if he does not like them, add more objects, save the video in the gallery or share his creation on a social network. The face buttons have a different behavior from the bubble buttons to show different types of edition that can be used in this concept. These kind of changes (made to all characters) are appealing when the user just wants to make quick changes on a video clip, not wasting much time.

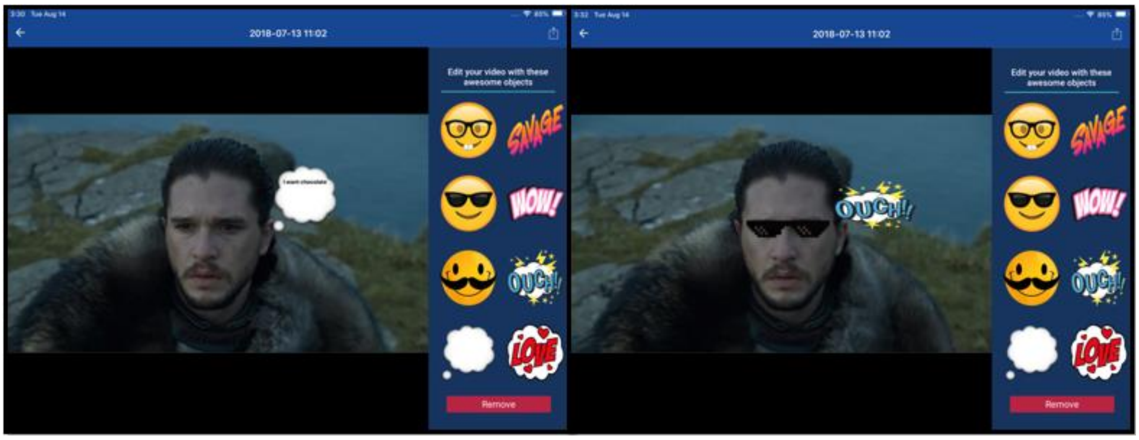


Figure 3.12 Computational Prototype – Editor view

3.3.2. Implementation and Architecture

There are three main algorithms that are very important for the development of this application: the algorithm used to make the facial recognition of the characters that appear on the video, the algorithm that makes the facial landmarks detection when the video is trimmed, so the user can add objects to the face of the characters and the algorithm that creates the new video with all the edition made by the users (Figure 3.13).

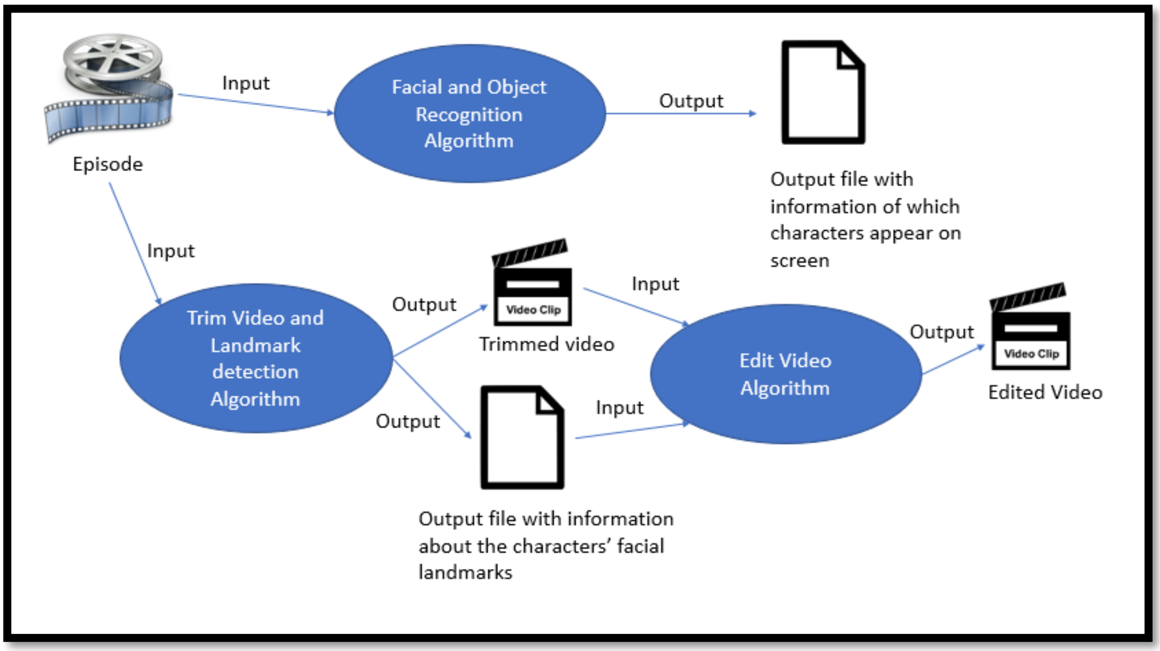

Episode
Input
Facial and Object Recognition Algorithm
Output
Output file with information of which characters appear on screen
Input
Trim Video and Landmark detection Algorithm
Output
Video Clip
Trimmed video
Input
Output
Output file with information about the characters' facial landmarks
Input
Edit Video Algorithm
Output
Video Clip
Edited Video

Figure 3.13 Implementation of the Algorithms

Facial and Object Recognition Algorithm

We started by developing the video and facial recognition components. The application has a JSON file (created by an auxiliary application, which the only objective is to create this file) with previously processed information about the location of faces and to whom they belong to. This information will be accessed while the user is viewing the video giving him the impression that it is made in real time. The recognition of the characters cannot be done in real time due to problems related to the performance of the application, but this will be explained in detail later. The algorithm that creates the file behaves as follows:

1. It creates an object of type **AVAssetReaderTrackOutput** [3], that defines an interface for reading media data, able to read the data coming from the video, frame by frame.
2. For all frames, it will fetch synchronously, media samples in the form of a **CMSampleBuffer** [5], an object that in addition to containing a frame of the video, also contains other information as the time in which that specific frame should appear.
3. Every 0.4 seconds, the buffer containing information about the fetched frame, is converted into a format that can be used to read as an image and facial detected.
    a. For each face found in this frame, facial recognition is performed in the area containing the face in question. If there are no faces in the frame, object recognition is performed on that image looking for special objects.
        i. If facial recognition gives a result with more than 80% of confidence, the location of the face, the time it appeared, and the id of the recognized person is kept in an array. If it is a special element, only the time and the name of the element is stored.

4. At the end, when there are no more frames, the array is converted to a JSON object and saved in a file.

This algorithm outputs a JSON file which is stored in the main application's resources. When the user enters the Video view (Figure 3.9), this file is converted into a data format array that can be read while the video is played. This data format is an object that contains information about a specific frame, such as the time that this frame will appear to the user, how many characters are on it and who they are. While viewing the video, if the facial recognition is on, for each frame that is loaded to the screen, an observer function will be triggered. This function, knowing the current time of the video, checks the array and sees if there are characters to be shown on this specific time. If a character is on that frame the information icon appears on the screen and if the character is making his first appearance, the recognition box also appears. If for some reason the character disappears, while showing the recognition box, and there is no information about him in the next position of the array, the box will disappear. This prevents delays on removing the box from the screen.

When we started experimenting, only face detection was tested, and the algorithm was run in real time. This algorithm performed facial detection in all frames but that caused a small delay showing the recognition box. To counter this delay, we searched for the perfect interval of time to make the detection and improve the algorithm performance, but without ruining the user experience. After some tests, it was concluded that detecting faces every 0.4 seconds was a good interval of time, since one face did not move enough to make the recognition box appear on a wrong location compared to the true movement of the face.

Now, that not all frames were being processed, the box started to appear without delays. After the facial recognition was implemented, we observed that it was taking too long to recognize the characters, especially in situations with many faces in the same frame. This was causing performance issues and with this in mind, a secondary application was created just to run this algorithm and create the file before viewing the video instead of trying to recognize elements in real time. Depending on the number of faces that appear on the frame, this algorithm can take between a little more than the original time of the video to nearly 1.5 times more. The part of detection and recognition is done using algorithms from Apple's **Vision** framework [7].

For the implementation of the facial recognition part, we had to create a machine learning model and transform it into a **CoreML** [6] model to use functions made available by Apple. Apple has already some machine learning models to identify a lot of objects but since what we intended was very specific, the models did not exist. To create this model, we used **Tensorflow** [91], a machine learning framework. We did some research on the characters we wanted, and we tried to put together a set of 100-200 photos from each character. Because some characters appeared less often, we had to get photographs of the actors in their daily life. We only used photos in which their appearance was similar to the one in the series.

To recognize the dragon, we used the **Inceptionv3** model, a model provided by Apple. It turned out that this model identified the dragon when it was flying as a vulture or a kite, if it was far away, and with its snout to the camera as a crocodile. Considering that in this series there are no such things, whenever one of these were identified, it was clear that the dragon was on the scene. Due to the dragon being a product of our fantasy and not a real animal, the confidence of the algorithm while detecting it, was not very high (this can also be seen by identifying it as different objects or animals). However, whenever the algorithm identified one of these elements described above, the dragon was always on the scene. Taking this into account, no matter the level of confidence, if one of these elements were identified, the algorithm assumes that the dragon is appearing.

In the series, we can also find giant wolves where one of them is white. We tested it with them too and the detector can even distinguish the white wolf from the others, in this case with more than 90% of confidence. This means that in case of special elements that do exist in the real world, it is possible to recognize them without difficulties. In the real world, a model should be trained for these cases. The problem here in training the model to recognize the dragon is the lack of photographs. The only way was using screenshots from the actual scene but that would be biased so we decided not to. The reason we chose the dragons instead of the wolves that can be recognized without flaws were thanks to the scenes that they appear. The wolves appear in darker scenarios and usually, violent ones, while the dragons appear in brighter scenarios and have some scenes without bloodshed. We considered that showing violence during the tests was not a good idea.

There is another file in our app's bundle resources. A JSON file that contains the information about the characters. This file was hand written and contains a **Dictionary** that links the names that were recognized with the information about each character. When the user taps the screen to watch extra information about a recognized character or special element, a function fetches this information from the dictionary.

Trim Video and Landmark Detection Algorithm

This algorithm is very similar to the face recognition one, having only some minor differences. This algorithm detects the facial landmarks of everyone that appears on the trimmed video and it runs when the user clicks to trim the video. First, the video is trimmed and saved inside the app's Document Directory. After that, a function runs the landmark detection algorithm that is similar to the facial recognition one but in this case, only face detection is performed. The output is also a JSON file.

The algorithm does the following:

1. First, it creates an object of type **AVAssetReaderTrackOutput** [3], that defines an interface for reading media data, to be able to read the data coming from the video, frame by frame.
2. For all the frames of the video, it fetches synchronously, media samples in the form of a **CMSampleBuffer** [5], an object that in addition to containing a frame of the video, also contains other information as the time in which that specific frame should appear.
3. Every 0.1 seconds, the buffer is converted into a format that can be used to read as an image and will be performed facial detection in that specific frame that was fetched. Nothing happens to the other frames. At the same time, a calculus is being made to update the progress bar using the time that belongs to the frame.
    a. If a face is found, the position of the landmarks and the time of their appearance is saved in an array.

4. At the end, when there are no more frames, the array is converted to a JSON object and saved in a file. This file is then saved in the document directory with the same name of the clip.

We need to know with a good precision where are the landmarks located, so we can put objects on top of a face or near it. The objects need to move with the character as smoothly as possible, so the ideal was to gather information related to all frames. But if the facial detection algorithm is used in all frames, it takes a considerable time to process all the video, so we needed to find the ideal time to make the transitions of an object from one frame to another as smooth as possible without making the user wait too long and without ruining the movement of the objects. After some experimentations, 0.1 seconds seemed like the perfect choice to achieve this.

At first, the video was trimmed during the viewing of the TV show, almost instantaneously, since the facial detection was not made during the viewing of the episode. The facial detection was only run after the selection of the video on the gallery. This also means that the output file with information about the faces in the trimmed video is only created in this part too. The problem with this approach was that the user had to wait some time, usually the duration of the clip, before doing some editing. With this in mind, the algorithm was moved to run during the viewing experience, so the user can continue watching the episode and does not need to wait some time later.

Edit Video Algorithm

This algorithm is responsible for overlaying the selected virtual objects to all the frames in the trimmed clip that is being edited and the creation of the new video. This algorithm will access the output file that is generated from the "Trim Video and Landmark Detection" algorithm to know where the faces are, in order to add objects to them. When the user enters the Editor view, a similar process to what happens when the user enters the Video view happens. The information is converted to a data format array to be used in the creation of the new video. This data format contains information about the location of the facial landmarks of each character and the time of their appearance. Since we have two types of objects, the face ones and the bubble ones, the algorithm needs to do an extra step when adding a bubble object to the frame. The general algorithm is shown next and then we talk about the differences between them.

1. First, the user selects the object that he wants to put in the video.
2. Then, the algorithm creates an object of type **AVAssetReaderTrackOutput** [3], that defines an interface for reading media data, to be able to read the data coming from the video, frame by frame.
3. All frames in the video are fetched synchronously. Media samples in the form of a **CMSampleBuffer** [5] are created, an object that in addition to containing a frame of the video, also contains other information as the time in which that specific frame should appear.
4. Then, for each frame, the object that the user selected is added to the frame according to the coordinates of a specific landmark related to the chosen object. For example, the moustache will be put on a mouth landmark. If the time from the array is the time of the frame that is being

edited, the object is put on the location presented in the array. If that time has already passed, while the algorithm is waiting for the next position in the array, the object is put on the location that it will have in the next 0.1 seconds (where the face will be).

    a. If the next location is very distant from the one where the object is being put now, the object is not shown.

5. These edited frames are all stored on an array.
6. Using **AVAssetWriter** [4], all the frames are re-encoded, and a new video is created with the all the changes made.

Although we only have information about the facial landmarks every 0.1 seconds, since it is the information present in the output file from the "Trim Video and Landmarks Detection" algorithm, we need to overlay the virtual object in all frames. We tested the option of using the "Trim Video and Landmarks Detection" algorithm to analyze all frames instead of only every 0.1 seconds, so we could add the objects to all frames in the correct position of the landmarks. We wanted to be sure that our prior decision to make the facial detection in intervals of time would not diminish the quality of the video that will be created. The results showed that when detecting all frames, landmarks could present some minor location changes from a frame to the next one while being detected, resulting in a behavior where the objects were always jumping. This was barely noticeable when moving the object based on the position that a landmark has every 0.1 seconds, like we are doing.

To ensure that the object is put in the right location, since we only have information about the landmarks every 0.1 seconds, when the time of the video matches the time of the frame that we have in the array, the object is put on the location that is specified in that position of the array. If the time has passed and the algorithm is waiting for the time of the video to match the next position of the array, the rest of the 0.1 seconds without information, the object is put already in its future position on all frames until the time matches again with the next position.

With this, the object is always a little bit ahead predicting the next place to be. This decision was made because otherwise, the object could be noted to follow the face instead of accompanying on some occasions. Since a face is not likely to make big changes in 0.1 seconds, this was not a problem and now it seems that the face and object are moving at the same time. The only problem that we had with our decision was with the disappearance of the face from the scene. Since we do not have information about all frames, the object would take more 0.1 seconds to disappear from the video than is supposed to, the time that is needed to receive new information and realize that the face is not there anymore. To prevent this from happening, if the change on the location of the object was bigger than a specific length on the next 0.1 seconds, the object would disappear, meaning that the characters had disappeared, and we are looking to new characters in a whole different place or even the same character but in a different scene.

Sometimes, a face cannot be detected due to problems with the light or bad positioning. This can happen only on the frame that was analyzed or during a very brief moment. In these special cases, if the next position of the array is 0.2 seconds in the future, which means that we do not have information about what is happening in the video for the next 0.2 seconds, since the other frame had no results (the one 0.1 seconds in the future), and a face in that position of the array is close to its current location, the object is shown. In any other case related with time, the object disappears, even if after one second there is a face in the same location.

This was all related to the general algorithm, but for objects that will behave like bubbles and are only expected to appear near one person at a specific time, some additional code had to be created.

When the user clicks on the object after defining its position, the object will be bound to that location and current time of the video. The size of the frames is different from the size of the player that is shown. This means that before the algorithm is run, calculus need to be made to know where the user intended to click exactly. After that, a function will access the array with the facial landmarks information and check how many characters appear in the frame that corresponds to that time of the video. Then, a search for the face that is nearer that specific point will be made and the object will be bound to a landmark of that face.

Now, the general algorithm runs as expected taking into account these coordinates. After that, the image will only appear on the frames that are in the specific time interval between the click and the duration of the animation (1,5 seconds or 3 seconds, depending if it is the thought balloon or not). These virtual objects that are supposed to be connected with only one face on a specific moment, brought a new problem. If the face disappears during this time interval, and in a rare case, another character appears on the exact same spot, the image will continue following this new character until the time ends.

To solve this problem, we had to ensure that if a face is not detected for 0.5 seconds, the animation is over. We chose this duration of time for the same reason already described above. Sometimes the face detector may have difficulty detecting some faces for reasons that we could not adjust. The time here is bigger because the object is not so face-dependent.

One problem that we could not solve is that the detector assumes that a face, that is on its side or in a bad angle, is faced front and completes the information about the landmark's position with fake information. The Apple's detector gives full information about a face or none at all. This may create some error with the objects but there is nothing we can do for now.

3.3.3. Evaluation

After the conclusion of the computer prototype, user tests were performed to evaluate its usability. The goal of these stage was to gather feedback about how the application was going to be used, so we could fix some flaws, make it more user friendly and to test our concept.

Participants and Methodology

This prototype was tested by 30 users with ages between 14 and 53 years old ($\bar{x}$= 26,7). Of the 30 testers, 8 were female.

The prototype was tested in more than one room, but always in a quiet place where the user could be at a table isolated from all distractions. During the execution of the test, no other possible tester was near the area where the test was being carried. The users had a scenario with some tasks to follow, available on a computer, but they had some freedom to use the prototype as they liked, as long as they tested all actions that are described on the tasks (Figure 3.14). The tasks were the same available in the paper prototype tests (Table 3.1).

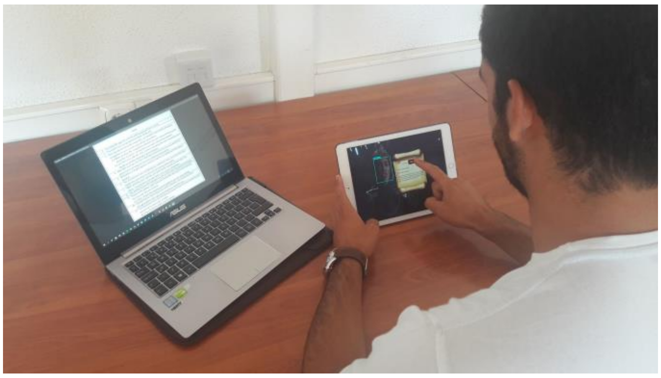


Figure 3.14 User test

Before the execution of the tests, the users had a brief explanation about the concept of TvTeller. It was also explained that some notes would be taken while they were using the application to ensure that all problems can be solved in a future phase. All tests were made using the same device, an iPad Air 2 and after that, all users filled out a questionnaire.

During these tests, the same video was available to all users. This video had enough to be able to test everything we need to. It was essential that in addition to faces, it was also inserted a non-human element, in which it was chosen to insert a moment with the dragon. All users were given complete freedom to explore the application and click wherever they wanted. This was mainly in the part of editing video clips where many of the users inserted a lot of assets to create a funny story.

After the testing, the users filled out a questionnaire. This questionnaire had 6 sections. The first section was about the user's gender and age. The second section contained questions about the user's habits related to our concept (Table 3.2).

Table 3.2 – Second section from the questionnaire

2. Series Habits
2.1. How often do you watch TV series?
2.2. If you watch series, what devices do you use to watch them?
2.3. Do you often watch videos created by other users as a form of entertainment? For example, in Instagram and Snapchat

The third section was about general aspects of the application, if it was difficult or not to perform a certain task. This section had statements to rate on a Likert scale from 1 (Very Hard) to 5 (Very Easy) where the user selects the number that is most suited to his experience (Table 3.3).

Table 3.3 Third section from the questionnaire

3. General Aspects
3.1. Activate Facial Recognition
3.2. Extract a clip from the video
3.3. Visualize information about the characters
3.4. Add objects to the clip that was extracted from the video
3.5. Share the clip on a social network

The fourth section focus on the usability and entertainment aspects. In this section the user also had to choose a grade from a Likert scale with the values 1 to 5 but in this one, 1 means "Strongly Disagree", and 5 means "Strongly Agree". This group has 9 questions but the last one needed previous information. So, the question 4.8 is a question with a "Yes" or "No" answer and 4.8.1 is only answered by users that say "Yes" on the previous one.

Table 3.4 Fourth Section from the questionnaire

4. Usability and Entertainment
4.1. Additional information about the characters is useful and complemented my experience
4.2. The appearance of the facial recognition box did not disturb my experience.
4.3. I can easily extract a video clip without losing focus
4.4 In a real context, I would use facial recognition functionality to watch my series
4.5. Probably, I will create new content
4.6. Probably, I will react to content created by others
4.7. Even if I do not watch the show, I can entertain myself with content that others created
4.8. Do you usually perform other tasks while watching a series?
4.8.1. If the answer to 4.8 is yes, the concept of TvTeller can help me following a series that is being broadcasted.

The fifth section contained some words that could describe the prototype, so the user could choose a maximum of 3 and the last section was an open answer where the users can give suggestions and comments.

Results and Discussion

Regarding the second section of the questionnaire, half of the participants (50%) watch series daily, 13,3% watch almost every day, 30% watch series on a weekly basis (some users stated that this is due to the series they watch only released one episode per week) and 6,7 % rarely watch series. The option of never watching was also available but no one selected it. The most used device to watch series is the computer selected by 83,3%. After the computer, the most used device is television with 70% followed by the smartphone with 36,7% and tablet with 6,7% (due to the fact that most of the testers did not possess a tablet). When users were asked if they are used to watch videos where other users added objects and stickers to the video, 90% said yes with the other 10% saying no.

In the third section, where questions about general aspects of the prototype are asked, users found that activating the facial recognition was easy with 73,3 % saying it was very easy.

Regarding extracting the clip from the episode, the results were also very positive as it was already expected, since that information was on the tutorial and the icon was also changed.

We also had the same results on watching information about a certain character (statement 3.3), with 70% finding it very easy and the other 30% finding it easy. Although in this case the results seem very good, almost everyone clicked during the time the box was showing. The people that only found it easy were the ones that did not click right away and had some doubts between clicking the information button and the character's face because there was not a box anymore. Almost all participants did not understand what the information button meant at first. Some users said that when the box disappears, without knowing the meaning of the information button, makes more difficult to them to know that they can click on the face. They said that the information button should also be explained in the tutorial.

Where users had more problems was in adding objects to the trimmed video (statement 3.4) where only 43,3 % found it very easy. Almost all users found it to be easy (46,7%). After that, 6,7% said that it was not easy or hard and 3,3% said that it was hard. This was due to the fact that they do not know how the buttons work, so the users did not understand the bubble objects initially and how to use them. Some users clicked on the button on a time where no face was present in the video (this case can be corrected with error messages) or at the end of its appearance and did not used the time slider after it to correct the situation. Others just clicked in a random part of the screen without linking the object to a face, so it can follow it. The other users found this to be very easy and did everything without any problem. Some users suggested that the face buttons' icon should be changed because the ones that are on the prototype gives the impression that the whole image will be added to the face and not only a moustache or a pair of glasses (Figure 3.12).

The last question of this section is about sharing the video on a social network with 86,7 % of the users finding it to be very easy to accomplish. Although this button is the one used for iOS, only 2 Android users complained that they did not know the symbol but since it was the only one available, they thought it was the right one. These results can be seen in the Figure 3.15.

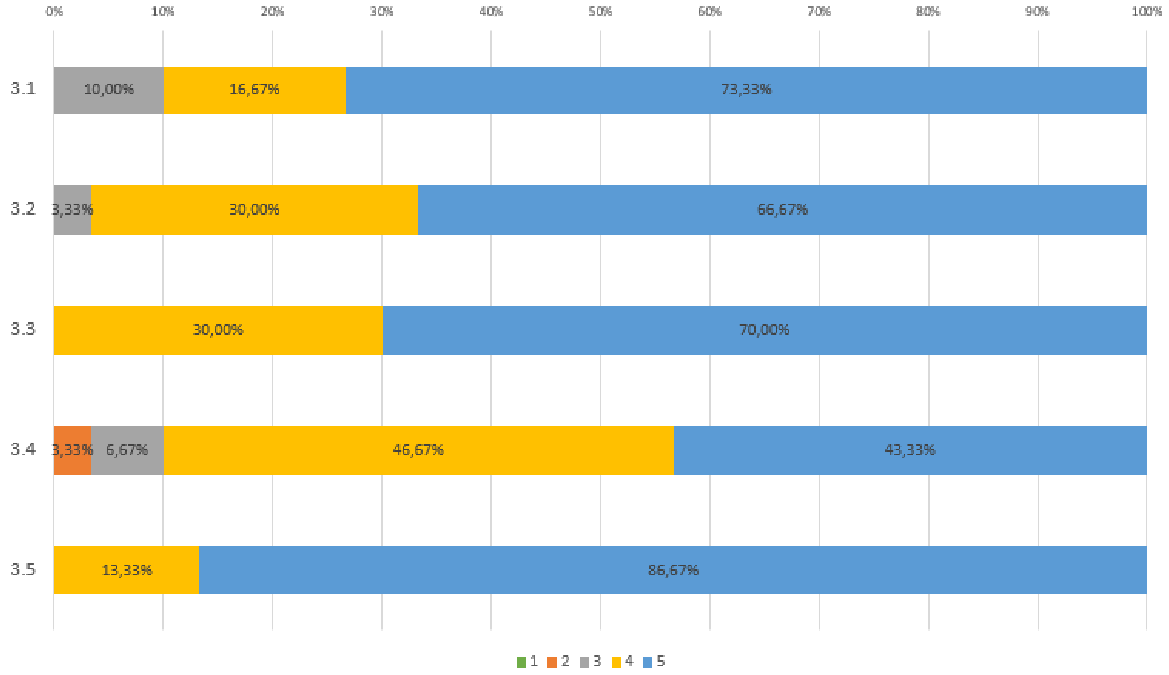


Figure 3.15 Third Section Results

After this section, that aimed to ascertain if users could use the prototype without difficulties, comes the section that aims to verify if the application does not ruin the viewing experience and to test the viability of or our concept (Figure 3.16).

As expected, the part where the user can get information about the characters was a success and not a single user found it to be difficult with 70% completely agreeing with the first statement.

Although we tried to put the box around the faces the minimum time possible, the results were not as good. Although the majority said that this did not ruined the experience, 20% did not agree.

When it came to extract the clip from the video without losing focus, half of the users (50%) completely agreed, 30% agreed, 13,3% did not agree or disagree and 6,7% disagreed. Although the results are satisfactory, they were lower than expected. The problem here was that some users were always clicking on the screen to see the progress bar instead of watching the video normally waiting for a notification. This may be due to the fact that they do not know about the notification that will appear at the end of the trimming on the first time. Other users also stated that they did not know how the trim would happen, if they had to choose the part to cut or the time of the trimmed moment.

When confronted with the question of whether they would use facial recognition in a real context to complement their experience (statement 4.4), only 40% completely agreed and 30% agreed. This was not expected but some of the users that gave an inferior value justified that they already know everyone that appears on the series that they watch. Although, in some special appearances they may resort to the facial recognition.

About the fact of creating content to share on a social network, the results were very dispersed (statement 4.5). This is normal, since some users like to share content and others prefer to watch. This can be seen here because when asked if they would react to content created by others (statement 4.6), the majority agreed with this affirmation with 36,67% completely agreeing and 46,67% agreeing.

The last 2 questions were more about proving the success of our concept. In 4.7, we asked the users if they can enjoy the videos even without knowing the series where half completely agreed and 33,3% agreed. This combined with the information that they would react to content created by others, proves that this part of the concept was well received.

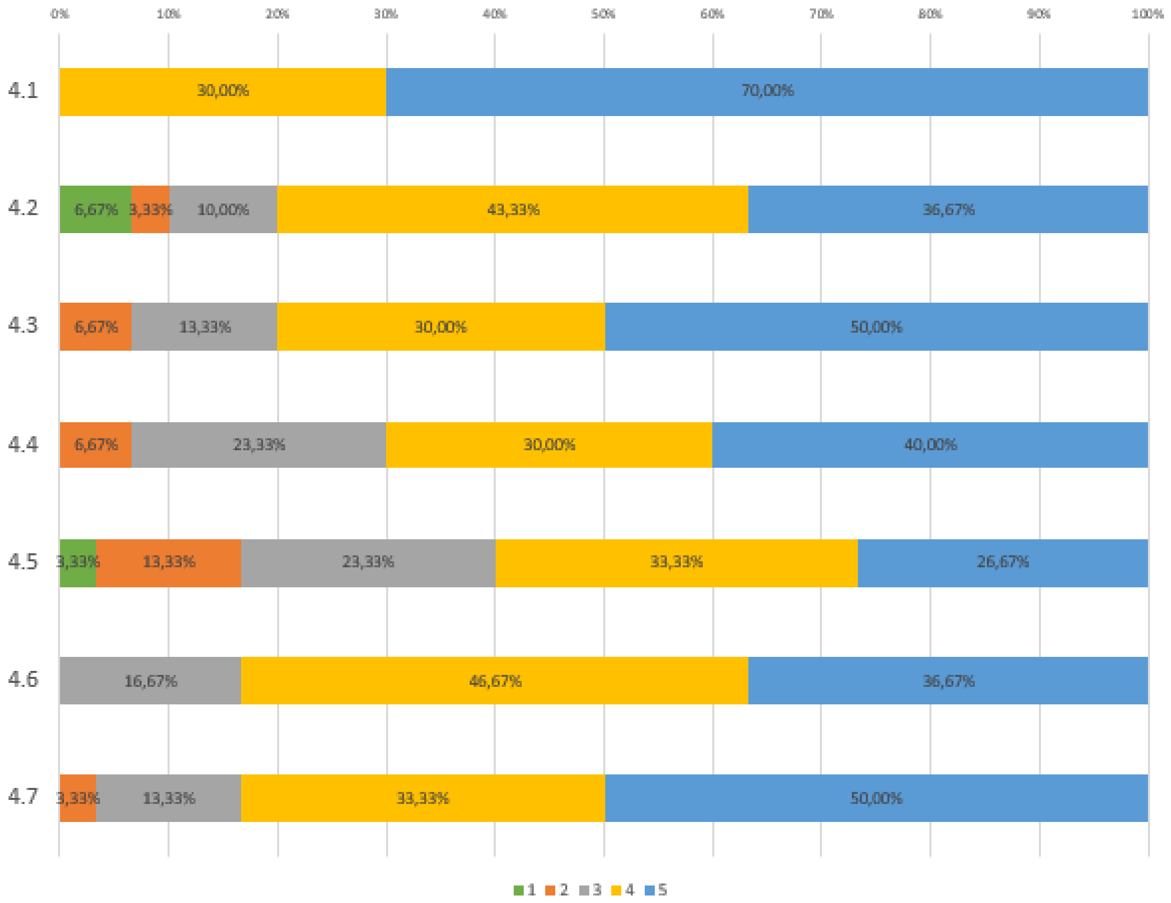


Figure 3.16 Fourth Section Results

For the last question, we needed to ensure that only users that use second screen, were answering to make the answer more reliable. In the question 4.8.1, when asked if this concept could help a user that lost some part of an episode to follow the series, of all the users that answered “Yes” in the 4.8 (50% of the users), 40% completely agreed and 40% agreed, meaning that 80% of the users gave a positive feedback.

4

Conclusions and Future work

The goal of this thesis was to develop and explore a new concept for interaction between a user and a television show using a tablet. We aimed to explore the possibility of turning the experience of watching a television show, into something not so passive and into a more social one. In order to evaluate the success of our concept, we created a prototype that gives the users the possibility to know more about their favorite series resorting to facial recognition and the possibility to create new content in the area of entertainment using augmented reality concepts and facial detection.

The development of the prototype had two iterations. During the first iteration, a paper prototype was developed. With this prototype, it was possible to better understand what we aimed for and what was our idea to present this concept. After the creation of our prototype, we made user tests, so we could evaluate what was its usability and user needs in order to discover what was needed to be changed. During the tests, the users were very cooperative and gave valuable feedback. This feedback was very important, since it allowed us to solve some usability problems in an earlier phase where the changes do not take much time and cost. In the second iteration, we implemented a computer prototype, taking into account all the feedback from the first one. After its implementation, this prototype was also submitted to user tests in order to find flaws and problems with the user experience that could not be caught during the tests in paper. These tests were done in a controlled environment. The feedback that we received from these tests was very important because it allowed us to find out what can be improved in a next iteration of this concept. All the users said that this application was relatively easy to use and very useful in the real world. Almost everyone that used it, said that it was a very good concept and were eager to use it on their daily routine while watching their favorite series.

In general, users loved all the features presented in this prototype but the one that they loved the most was the possibility to add thought balloons and the other bubble objects. The part related to adding objects to a face, such as the moustache, is something that they are already used to, using face filters on applications like Instagram or Snapchat, and although they loved the possibility to add them to TV characters, what they loved the most were the other objects stated above since it is something fresh, that they are not used to see in an application.

Almost everyone loved the entertainment part and said that it is amazing having the possibility to create that kind of content using clips from series and in such an easy way. The feature where the users can watch information about the characters was also very well received. Some users also shared that this feature solves some of their problems while watching a TV series, since they had the necessity to know more about the characters while they were watching it.

All this feedback made us conclude that this concept was very well received, but some problems need to be solved in the future. Some issues with the prototype were already solved, for example the face objects' buttons received a new icon and the app's tutorial was updated. Apple's facial detection algorithm does not give information about the face's angle and this can lead to problems with some of the objects added to the clip. Very recently, Apple improved some of its libraries where now exists in fact information about this. The problem is that it is not very accurate yet, giving information that is very limited and not always correct. Other thing that can be done to solve this problem is trying other face detector.

In the future we could try to take this concept even further and identify the facial expression of the characters to create new animations based on emotions. Other thing that can be done is taking the entertainment part even further and find a way to add some kind of sound interactions or changing the face of characters instead of adding objects to them, like for example adding the user's mouth on top of the character's mouth while saying something to pretend that the character is saying something that in fact is said by the user.

5

Bibliography

[1] Ababsa, F., Maidi, M., Didier, J.Y, & Mallem, M. (2008). Vision-based tracking for mobile augmented reality. Multimedia Services in Intelligent Environments, 297-326.

[2] Andersen, R. E., Crespo, C. J., Bartlett, S. J., Cheskin, L. J., & Pratt, M. (1998). Relationship of physical activity and television watching with body weight and level of fatness among children: results from the Third National Health and Nutrition Examination Survey. Jama, 279(12), 938-942.

[3] Apple. AVAssetReaderTrackOutput. Available at: https://developer.apple.com/documentation/avfoundation/avassetreadertrackoutput . Visited in August 20, 2018.

[4] Apple. AVAssetWriter. Available at: https://developer.apple.com/documentation/avfoundation/avassetwriter . Visited in August 20, 2018.

[5] Apple. CMSampleBuffer. Available at: https://developer.apple.com/documentation/coremedia/cmsamplebuffer-u71 . Visited in August 20, 2018.

[6] Apple. CoreML. Available at: https://developer.apple.com/documentation/coreml . Visited in August 20, 2018.

[7] Apple. Vision. Available at: https://developer.apple.com/documentation/vision . Visited in August 20, 2018.

[8] Arya, S., Mount, D. M., Netanyahu, N. S., Silverman, R., & Wu, A. Y. (1998). An optimal algorithm for approximate nearest neighbor searching fixed dimensions. Journal of the ACM (JACM), 45(6), 891-923.

[9] atelevisão. Já está disponibilizada a aplicação interativa do «Agarra a Música». Available at: https://www.atelevisao.com/sic/ja-esta-disponibilizada-aplicacao-interativa-do-agarra-musica/ . Visited in January 31, 2018

[10] Azuma, R.T., Hoff, B.R, Neely III, H.E., Sarfaty, R., Daily, M.J, Bishop, G., ... & Nichols, R. (1998, November). Making augmented reality work outdoors requires hybrid tracking. In Proceedings of the First International Workshop on Augmented Reality (Vol. 1).

[11] Bachmayer, S., Lugmayr, A., & Kotsis, G. (2009, December). New social & collaborative interactive TV program formats. In Proceedings of the 7th International Conference on Advances in Mobile Computing and Multimedia (pp. 121-128). ACM.

[12] Billinghurst, M., Clark, A., & Lee, G. (2015). A survey of augmented reality. Foundations and Trends® in Human–Computer Interaction, 8(2-3), 73-272.

[13] Blomgren, S., & Hertz, M. (2015). Facing the differences between Facebook and OpenCV: A facial detection comparison between Open Library Computer Vision and Facebook.

[14] Brunelli, R., & Poggio, T. (1993). Face recognition: Features versus templates. IEEE transactions on pattern analysis and machine intelligence, 15(10), 1042-1052.

[15] Cambridge Dictionary. Livestream. Available at: https://dictionary.cambridge.org/dictionary/english/livestream . Visited in February 6, 2018.

[16] Cambridge Dictionary. Meaning of "television" in the English Dictionary. Available at: https://dictionary.cambridge.org/dictionary/english/television . Visited in February 2 ,2018.

[17] Cesar P., Bulterman D.C.A, Jansen A.J. (2008) Usages of the Secondary Screen in an Interactive Television Environment: Control, Enrich, Share, and Transfer Television Content. In: Tscheligi M., Obrist M., Lugmavr A. (eds) Changing Television Environments. EuroITV 2008. Lecture Notes in Computer Science, vol 5066. Springer, Berlin, Heidelberg.

[18] Comaniciu, D., Ramesh, V., & Meer, P. (2000). Real-time tracking of non-rigid objects using mean shift. In Computer Vision and Pattern Recognition, 2000. Proceedings, IEEE Conference on (Vol. 2, pp. 142-149). IEEE.

[19] Comport, A. I., Marchand, É., & Chaumette, F. (2003, October). A real-time tracker for markerless augmented reality. In Proceedings of the 2nd IEEE/ACM International Symposium on Mixed and Augmented Reality (p. 36). IEEE Computer Society.

[20] Datar, M., Immorlica, N., Indyk, P., & Mirrokni, V. S. (2004, June). Locality-sensitive hashing scheme based on p-stable distributions. In Proceedings of the twentieth annual symposium on Computational geometry (pp. 253-262). ACM.

[21] D'heer, E., Courtois, C., & Paulussen, S. (2012, July). Everyday life in (front of) the screen: the consumption of multiple screen technologies in the living room context. In Proceedings of the 10th European conference on Interactive tv and video (pp. 195-198). ACM.

[22] Digiday. WTF is OTT?. Available at: https://digiday.com/media/what-is-over-the-top-ott/ . Visited in February 8, 2018.

[23] Dorward, L. J., Mittermeier, J. C., Sandbrook, C., & Spooner, F. (2017). Pokémon Go: Benefits, costs, and lessons for the conservation movement. Conservation Letters, 10(1), 160-165.

[24] Edwards, G. J., Taylor, C. J., & Cootes, T. F. (1998, April). Learning to indentify and track faces in image sequences. In Automatic Face and Gesture Recognition, 1998. Proceedings. Third IEEE International Conference on (pp. 260-265). IEEE.

[25] Feiner, S., MacIntyre, B., Höllerer, T., & Webster, A. (1997). A touring machine: Prototyping 3D mobile augmented reality systems for exploring the urban environment. Personal Technologies, 1(4), 208-217.

[26] Foxlin, E., Altshuler, Y., Naimark, L., & Harrington, M. (2004, November). Flighttracker: A novel optical/inertial tracker for cockpit enhanced vision. In Proceedings of the 3rd IEEE/ACM International Symposium on Mixed and Augmented Reality (pp. 212-221). IEEE Computer Society.

[27] Foxtail Marketing. Augmented Reality Marketing. Available at : https://foxtailmarketing.com/business-can-benefit-augmented-reality-marketing/augmented-reality-main/ . Visited in February 10, 2018.

[28] Gerbner, G., Gross, L., Morgan, M., & Signorielli, N. (1986). Living with television: The dynamics of the cultivation process. Perspectives on media effects, 1986, 17-40.

[29] Glick, I. O., & Levy, S. J. (2017). Living with television. Transaction Publishers.

[30] Gray, J. (2009). Television entertainment. Routledge.

[31] Hagbi, N., Bergig, O., El-Sana, J., & Billinghurst, M. (2011). Shape recognition and pose estimation for mobile augmented reality. IEEE transactions on visualization and computer graphics, 17(10), 1369-1379.

[32] Haimson, O. L., & Tang, J. C. (2017, May). What makes live events engaging on Facebook Live, Periscope, and Snapchat. In Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems (pp. 48-60). ACM.

[33] Hamilton, W. A., Garretson, O., & Kerne, A. (2014, April). Streaming on twitch: fostering participatory communities of play within live mixed media. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (pp. 1315-1324). ACM.

[34] Hjelmås, E., & Low, B. K. (2001). Face detection: A survey. Computer vision and image understanding, 83(3), 236-274.

[35] Höllerer, T., & Feiner, S. (2004). Mobile augmented reality. Telegeoinformatics: Location-Based Computing and Services. Taylor and Francis Books Ltd, London, UK, 21.

[36] Huang, F. J., & Chen, T. (2000). Tracking of multiple faces for human-computer interfaces and virtual environments. In Multimedia and Expo, 2000. ICME 2000. 2000 IEEE International Conference on (Vol. 3, pp. 1563-1566). IEEE.

[37] Huston, A. C. (1992). Big world, small screen: The role of television in American society. U of Nebraska Press.

[38] Iyengar, S., & Kinder, D. R. (2010). News that matters: Television and American opinion. University of Chicago Press.

[39] Jafri, R., & Arabnia, H. R. (2009). A survey of face recognition techniques. Jips, 5(2), 41-68.

[40] Katiyar, A., Kalra, K., & Garg, C. (2015). Marker Based Augmented Reality. Advances in Computer Science and Information Technology (ACSIT), 2(5), 441-445.

[41] Kaytoue, M., Silva, A., Cerf, L., Meira Jr, W., & Raïssi, C. (2012, April). Watch me playing, i am a professional: a first study on video game live streaming. In Proceedings of the 21st International Conference on World Wide Web (pp. 1181-1188). ACM.

[42] Kim, S. C., Koo, H. S., Kim, H., & Cheong, J. (2016, December). Implementation of AR-Based Hybrid Broadcasting System by TV Viewer's Preferred Content Provider. In Information Science and Security (ICISS), 2016 International Conference on (pp. 1-4). IEEE.

[43] Kraemer, J., & Wohlfarth, M. (2015). Regulating over-the-top service providers in two-sided content markets: Insights from the Economic Literature.

[44] Li, Y., Goshtasby, A., & Garcia, O. (2000). Detecting and tracking human faces in videos. In Pattern Recognition. 2000. Proceedings. 15th International Conference on (Vol. 1, pp.807-810). ieee.

[45] Liu, Z., & Wang, Y. (2000). Face detection and tracking in video using dynamic programming. In Image Processing, 2000. Proceedings. 2000 International Conference on (Vol. 1, pp.53-56). IEEE.

[46] Looser, J., Billinghurst, M., & Cockburn, A. (2004, June). Through the looking glass as an interface tool for Augmented Reality interfaces. In Proceedings of the 2nd international conference on Computer graphics and interactive techniques in Australasia and South East Asia (pp. 204-211). ACM

[47] Lowe, D. G. (2004). Distinctive image features from scale-invariant keypoints. International journal of computer vision, 60(2), 91-110.

[48] Lu, K. Y. (2005). Interaction design principles for interactive television (Doctoral dissertation, Georgia Institute of Technology).

[49] Lu, S. Y., Shpitalni, M., & Gadh, R. (1999). Virtual and augmented reality technologies for product realization. CIRP Annals-Manufacturing Technology, 48(2), 471-495.

[50] Lv, Z., Halawani, A., Feng, S., Ur Réhman, S., & Li, H. (2015). Touch-less interactive augmented reality game on vision-based wearable device. Personal and Ubiquitous Computing, 19(3-4), 551-567.

[51] Madden, L. (2011). Professional augmented reality browsers for smartphones: programming for junaio, lavar and wikitude. John Wiley & Sons.

[52] Manjunath, B. S., Chellappa, R., & von der Malsburg, C. (1992, June). A feature based approach to face recognition. In Computer Vision and Pattern Recognition, 1992. Proceedings CVPR'92., 1992 IEEE Computer Society Conference on (pp. 373-378). IEEE.

[53] Marcus, A., Roibás, A. C., & Sala, R. (Eds.). (2009). Mobile TV: Customizing Content and Experience. Springer Science & Business Media.

[54] Marvel. Available at: https://marvelapp.com/ . Visited in August 7, 2018.

[55] Mason, S. (2006). Mobile TV–Results from the BT Movio DAB-IP trial in Oxford. EBU Technical Review.

[56] Meme. You're in trouble, enemies are approaching. What to do? . Available at : https://me.me/i/ig-daenerys-stormbormkhaleesi-youre-in-trouble-enemies-are-approaching-what-14342699 . Visited in February 10, 2018.

[57] Milgram, P., Takemura, H., Utsumi, A., & Kishino, F. (1995, December). Augmented reality: A class of displays on the reality-virtuality continuum. In Telemanipulator and telepresence technologies (Vol. 2351, pp. 282-293). International Society for Optics and Photonics.

[58] Miller, M. K., Tang, J. C., Venolia, G., Wilkinson, G., & Inkpen, K. (2017, May). Conversational Chat Circles: Being All Here Without Having to Hear It All. In Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems (pp. 2394-2404). ACM.

[59] Moreno, P., Bernardino, A., & Santos-Victor, J. (2009). Improving the SIFT descriptor with smooth derivative filters. Pattern Recognition Letters, 30(1), 18-26.

[60] Nakai, T., Kise, K., & Iwamura, M. (2006). Use of affine invariants in locally likely arrangement hashing for camera-based document image retrieval. Document Analysis Systems VII, 541-552.

[61] Nassani, A., Kim, H., Lee, G., Billinghurst, M., Langlotz, T., & Lindeman, R. W. (2016, November). Augmented reality annotation for social video sharing. In SIGGRAPH ASIA 2016 Mobile Graphics and Interactive Applications (p. 9). ACM.

[62] Nielsen Company. (2013). Action figures: How second screens are transforming TV viewing. Available in: http://www.nielsen.com/us/en/insights/news/2013/action-figures--how-second-screens-are-transforming-tv-viewing.html . Accessed on February 1, 2018.

[63] Olsen, D. R., Bunn, D., Boulter, T., & Walz, R. (2012). Interactive television news. ACM Transactions on Multimedia Computing, Communications and Applications (TOMM), 8(2), 19.

[64] Pires, K., & Simon, G. (2015, March). YouTube live and Twitch: a tour of user-generated live streaming systems. In Proceedings of the 6th ACM Multimedia Systems Conference (pp. 225-230). ACM.

[65] Pope, A.R., & Lowe, D. G. (1993, May). Learning object recognition models from images. In Computer Vision, 1993. Proceedings., Fourth International Conference on (pp. 296-301). IEEE.

[66] Ramkumar, G., & Manikandan, M. (2013). Face recognition-survey. Science publication .org, 260-268.

[67] Rohs, M. (2007). Marker-based embodied interaction for handheld augmented reality games. Journal of Virtual Reality and Broadcasting, 4(5), 1860-2037.

[68] Rowley, H. A., Baluja, S., & Kanade, T. (1998). Neural network-based face detection. IEEE Transactions on pattern analysis and machine intelligence, 20(1), 23-38.

[69] Sánchez-Acevedo, M.A., Sabino-Moxo, B.A., & Márquez-Dominguez, J.A. (2017). Mobile Augmented Reality. Mobile Platforms, Design, and Apps for Social Commerce, 153.

[70] Schatz, R., Wagner, S., Egger, S., & Jordan, N. (2007, June). Mobile TV becomes social-integrating content with communications. In Information Technology Interfaces, 2007. ITI 2007. 29th International Conference on (pp. 263-270). IEEE.

[71] Schmalstieg, D., & Hollerer, T. (2016). Augmented reality: principles and practice. Addison-Wesley Professional.

[72] Schmalstieg, D., & Wagner, D. (2009). Mobile phones as a platform for augmented reality. connections, 1, 3.

[73] Singer, D. G., & Singer, J. L. (2012). Handbook of Children and the Media. Sage.

[74] Sjöblom, M., & Hamari, J. (2017). Why do people watch others play video games? An empirical study on the motivations of Twitch users. Computers in Human Behavior, 75, 985-996.

[75] Smith, T., Obrist, M., & Wright, P. (2013, June). Live-streaming changes the (video) game. In Proceedings of the 11th european conference on Interactive TV and video (pp. 131-138). ACM.

[76] Södergård, C. (2003). Mobile television–technology and user experiences. Report on the Mobile-TV project. VTT, Finland.

[77] Statista. Consumer mobile device augmented reality applications(embedded/standalone) worldwide from 2016 to 2022(in millions) *. Available at: https://www.statista.com/statistics/591181/global-augmented-virtual-reality-market-size/. Accessed on February, 1, 2018.

[78] Statista. Number of mobile phone users worldwide from 2013 to 2019 (in billions). Available at: https://www.statista.com/statistics/274774/forecast-of-mobile-phone-users-worldwide/. Accessed on January, 28, 2018.

[79] Statista. Number of mobile phone video viewers in the United States from 2014 to 2020 (in millions) .Available in: https://www.statista.com/statistics/209348/mobile-video-viewers-in-the-united-states/ . Visited in August 7, 2018.

[80] Statista. Over-the-top (OTT) revenue worldwide from 2010 to 2022. Available at: https://www.statista.com/statistics/260179/over-the-top-revenue-worldwide/. Visited in February 7, 2018.

[81] Statista. Smartphones Beat Tv for young adults in the U.S. Available in: https://www.statista.com/chart/8660/smartphone-vs-tv-usage/ . Accessed on January 31, 2018.

[82] Strangelove, M. (2015). Post-TV: Piracy, cord-cutting, and the future of television. University of Toronto Press.

[83] Subrahmanyam, K., Reich, S. M., Waechter, N., & Espinoza, G. (2008). Online and offline social networks: Use of social networking sites by emerging adults. Journal of applied developmental psychology, 29(6), 420-433.

[84] Sugie, M., Gohshi, S., Takeshita, H., & Mori, C. (2014, December). Subjective assessment of super-resolution 4K video using paired comparison. In Intelligent Signal Processing and Communication Systems (ISPACS), 2014 International Symposium on (pp. 042-047). IEEE.

[85] Sutherland, I. E. (1965). The ultimate display. Multimedia: From Wagner to virtual reality, 506-508.

[86] Sutherland, I. E. (1968, December). A head-mounted three dimensional display. In Proceedings of the December 9-11, 1968, fall joint computer conference, part I (pp. 757-764). ACM.

[87] Tang, J. C., Kivran-Swaine, F., Inkpen, K., & Van House, N. (2017, February). Perspectives on Live Streaming: Apps, Users, and Research. In Companion of the 2017 ACM Conference on Computer Supported Cooperative Work and Social Computing (pp. 123-126). ACM.

[88] Tang, J. C., Venolia, G., & Inkpen, K. M. (2016, May). Meerkat and periscope: I stream, you stream, apps stream for live streams. In Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems (pp. 4770-4780). ACM.

[89] Tang, J., Venolia, G., Inkpen, K., Parker, C., Gruen, R., & Pelton, A. (2017, December). Crowdcasting: Remotely Participating in Live Events Through Multiple Live Streams. Proc. ACM Hum.-Comput. Interact. 1, CSCW, Article 98 (December 2017), 18 pages. DOI: https://doi.org/10.1145/3134733

[90] Tannenbaum, P. H. (2014). The entertainment functions of television. Psychology Press.

[91] Tensorflow. Available at: https://www.tensorflow.org/. Visited in August 20, 2018.

[92] Turk, M., & Pentland, A. (1991). Eigenfaces for recognition. Journal of cognitive neuroscience. 3(1), 71-86.

[93] Uchiyama, H., & Marchand, E. (2011, October). Toward augmenting everything: Detecting and tracking geometrical features on planar objects. In Mixed and Augmented Reality (ISMAR), 2011 10th IEEE International Symposium on (pp. 17-25). IEEE.

[94] Uchiyama, H., & Marchand, E. (2012, February). Object detection and pose tracking for augmented reality: Recent approaches. In 18th Korea-Japan Joint Workshop on Frontiers of Computer Vision (FCV).

[95] Variety. How HBO, CBS Seek Upper Hand in Pay-TV Deals with OTT Moves. Available at : http://variety.com/2014/digital/news/how-hbo-cbs-seek-upper-hand-in-pay-tv-deals-with-ott-moves-1201332667/ . Visited in February 8, 2018.

[96] Vezhnevets, V., Sazonov, V., & Andreeva, A. (2003, September). A survey on pixel-based skin color detection techniques. In Proc. Graphicon (Vol. 3, pp. 85-92).

[97] vizrt. NRK's augmented reality graphics World Cup studio. Available in: http://www.vizrt.com/casestudies/44997/NRKs_augmented_reality_graphics_World_Cup_studio . Visited in February 2, 2018.

[98] Wang, H., Wang, Y., & Cao, Y. (2009). Video-based face recognition: A survey. World Academy of Science, Engineering and Technology, International Journal of Computer, Electrical, Automation, Control and Information Engineering, 3(12), 2809-2818.

[99] We Are Tv. Available at: https://wearetv.com/ .Visited in January 31, 2018.

[100] Wei, R. (2008). Motivations for using the mobile phone for mass communications and entertainment. Telematics and Informatics, 25(1), 36-46.

[101] Wethington, H., Pan, L., & Sherry, B. (2013). The Association of Screen Time, Television in the Bedroom, and Obesity Among School-Aged Youth: 2007 National Survey of Children's Health. Journal of School Health, 83(8), 573-581.

[102] Whiting, A., & Williams, D. (2013). Why people use social media: a uses and gratifications approach. Qualitative Market Research: An International Journal, 16(4), 362-369.

[103] Yang, M. H., Kriegman, D. J., & Ahuja, N. (2002). Detecting faces in images: A survey. IEEE Transactions on pattern analysis and machine intelligence, 24(1), 34-58.

[104] You, S., Neumann, U., & Azuma, R. (1999, March). Hybrid inertial and vision tracking for augmented reality registration. In Virtual Reality, 1999. Proceedings., IEEE (pp. 260-267). IEEE.

[105] Yourstory. Virtual Reality 101: An Introduction to VR. Available at: https://yourstory.com/2016/06/virtual-reality-introduction/ . Visited in February 10, 2018.

[106] Yuille, A. L., Hallinan, P. W., & Cohen, D. S. (1992). Feature extraction from faces using deformable templates. International journal of computer vision, 8(2), 99-111.

[107] Zhang, C., & Zhang, Z. (2010). A survey of recent advances in face detection.

[108] Zhang, G., Patuwo, B. E., & Hu, M. Y. (1998). Forecasting with artificial neural networks:: The state of the art. International journal of forecasting, 14(1), 35-62.

[109] Zhao, W., Chellapa, R., Phillips, P. J., & Rosenfeld, A. (2003). Face recognition: A literature survey. ACM computing surveys (CSUR), 35(4), 399-458.

[110] Zhou, F., Duh, H. B. L., & Billinghurst, M. (2008, September). Trends in augmented reality tracking, interaction and display: A review of ten years of ISMAR. In Proceedings of the 7th IEEE/ACM International Symposium on Mixed and Augmented Reality (pp. 193-202). IEEE Computer Society.

[111] Zhou, S., Krueger, V., & Chellapa, R. (2002, May). Face recognition from video: A condensation approach. In Automatic Face and Gesture Recognition, 2002. Proceedings. Fifth IEEE International Conference on (pp. 221-226). IEEE.

[112] Zillmann, D., & Vorderer, P. (Eds.). (2000). Media entertainment: The psychology of its appeal. Routledge.