Deliverable

Project Acronym:	**ImmersiaTV**
Grant Agreement number:	688619
Project Title:	*Immersive Experiences around TV, an integrated toolset for the production and distribution of immersive and interactive content across devices.*

D4.3 Technical Evaluation

Revision: 0.3

Authors:

Saeed Mahmoudpour (imec)

Adriaan Barri (imec)

Delivery date: M11

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement 688619		
Dissemination Level		
P	Public	x
C	Confidential, only for members of the consortium and the Commission Services	

Abstract: This deliverable will detail the technical aspects of the outcome of the pilots, what worked and what didn't, what should be improved from a technical performance perspective. The technical evaluation will be focused on the performance monitoring of different components delivered in the iterations of the project, the lessons learned in each iteration, and guidelines to improve performance.

REVISION HISTORY

Revision	Date	Author	Organisation	Description
0.1	13/12/2016	Saeed Mahmoudpour/ Adriaan Barri	imec	First draft version of the deliverable
0.2	30/01/2016	Saeed Mahmoudpour/ Adriaan Barri	imec	First review
0.3	08/03/2017	Saeed Mahmoudpour/ Adriaan Barri	imec	Second review

EXECUTIVE SUMMARY

This deliverable describes the detailed technical evaluation of the outcome of Pilot 1 in the ImmersiaTV project. The aim of the document is to report the technical specifications and performance of the project components, describe the current issues and provide guidelines for better performance.

The first section provides an overview of the ImmersiaTV platform and components. In section 2, we go through the most noteworthy technical problems and limitations faced in Pilot 1 at both the production side and the end-user side: In video capturing and visualization observed optical distortions are mentioned and described. Next, the video stitching software used in Pilot 1 is introduced and the stitching errors are further explained. In the video editing part, we reported the improvements in Adobe Premiere plug-in, portals and scene transitions. The distortion types observed in Pilot 1 (including Blocking, Ringing, blur and colour artifacts) are described and shown by examples. As well, the issues related to distribution, reception and interaction are described in detail.

According to the observed issues and lessons learned in Pilot 1, a number of guidelines are provided in section 4 to improve the performance of pre-production and post-production. Moreover, some desirable features are suggested that could be added to the current ImmersiaTV platform. Section 5 summarizes the current status of user requirements mentioned in deliverable D 2.3 and section 6 addresses the conclusion.

CONTRIBUTORS

First Name	Last Name	Company	e-Mail
Sergi	Fernandez	i2CAT	Sergi.fernandez@i2cat.net
Joan	Llobera	i2CAT	Joan.llobera@i2cat.net
Isaac	Fraile	i2CAT	Isaac.fraile@i2cat.net
David	Gomez	i2CAT	David.gomez@i2cat.net
Stephane	Valente	VideoStitch	Stephane.valente@video-stitch.com
Szymon	Malewski	PSNC	szymonm@man.poznan.pl

CONTENTS

TABLE OF FIGURES

LIST OF TABLES

LIST OF ACRONYMS

Acronym	Description
VR	Virtual Reality
HMD	Head Mounted Display
QoE	Quality of Experience
DASH	Dynamic Adaptive Streaming over HTTP
UHD	Ultra High Definition
HDMI	High-Definition Multimedia Interface
GoP	Group of Pictures
DoP	Director of Photography
HDRI	High Dynamic Range Imaging

1. INTRODUCTION

1.1. Purpose of this document

This report will detail, iteratively, the technical aspects of the outcome of the pilots: what worked and what didn't, what should be improved from a technical performance perspective. The technical evaluation will be centred on the performance monitoring of the components delivered in the various iterations of the project.

1.2. Scope of this document

The remainder of this document in its present form is divided into three sections. It starts with a brief overview of the ImmersiaTV platform and the technical specifications of video capture, stitching, editing, compression, distribution, reception & interaction, and QoE feedback. A more elaborate description can be found in deliverable D3.1 (Architecture design). Section 3 goes through the most noteworthy technical problems and limitations faced in Pilot 1 at both the production side and the end-user side. The impact of the occurred technical problems on the user experience will be reported in deliverable D4.4 (User Evaluation). Section 4 lists the most important lessons learned and some guidelines gathered from ImmersiaTV partners.

1.3. Relation with other ImmersiaTV activities

This deliverable is part of task T4.3 (User and Technical Evaluation) in WP4 (Demonstration Pilots). The relationship between this task and the other WP tasks is shown below.

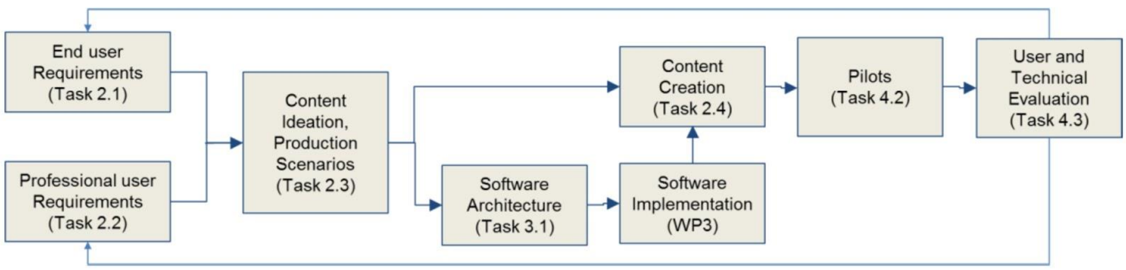

Figure 1: Role of T4.3 within the ImmersiaTV platform

2. IMMERSIATV PLATFORM OVERVIEW

ImmersiaTV aims to distribute omnidirectional and directive audiovisual content simultaneously to head mounted displays (HMD), companion screens and the traditional TV. The content distributed is constituted of one or more omnidirectional videos, complemented with several directive shots, and metadata detailing how to merge these streams in an immersive display, in coordination with directive and omnidirectional videos also shown in traditional TVs and tablets.

As detailed in Deliverable D3.1 - Architecture Design[1], the ImmersiaTV platform involves the following processes:

2.1. Video Capture

The acquisition of the video streams coming from 360 omnidirectional camera systems as well as other sources such as high-resolution directive cameras, video clips, textual information and other metadata required for generating omnidirectional video enriched with audiovisual and auxiliary information in further stages.

2.2. Video Stitching

The combination of video streams from the omnidirectional camera systems (constructed using multiple physical cameras) into a 360-degree video.

2.3. Post-Production (Offline or Live)

A set of tools and plugins for the montage, colour grading, and processing/enhancement of the 360-degree video material. The outcome of the post-production phase is a set of synchronized video signals to be played on the each of the display devices of the end user (HMD, companion screen, and traditional TV).

2.4. Compression (Encoding)

Reduction of the bitrate of the video signals (produced offline or streamed in real-time) by removing unnecessary or less important information.

2.5. Distribution

Transmission of the encoded video signals from the server at the production side to the centralized computing unit at the end-user side. The transmission channel encapsulates the selected video streams into network protocols.

2.6. Reception & Interaction

Reception of the encoded video signals at the end-user side. The player communicates with different display devices to make sure the video signals are properly received and synchronized.

2.7. QoE feedback

The QoE module estimates the perceptual quality of the visualization on the display devices at the end-user side by means of subjective and objective metrics and suggests adjustments of the network/encoding/rendering parameters at regular time intervals that will optimize the overall quality of experience of the end user.

[1] http://www.immersiatv.eu/wp-content/uploads/2015/11/D3.1-Architecture-design-version-0.9.pdf

3. TECHNICAL SPECIFICATIONS OF PILOT 1 AND THE OBSERVED ISSUES

Pilot 1 focuses on offline video production scenarios. The stitching of the omnidirectional scenes captured by several cameras was performed using AutoPano. Omnidirectional and directive streams are processed and aligned using off the shelf stitching tools and an Adobe Premiere Plugin specifically designed to edit multi-platform videos, combining omnidirectional and traditional videos, both standalone and embedded as video inserts within the omnidirectional scene.

After offline production, the video signals are compressed using the H.264 codec, transferred to a DASH server, and transmitted to central computing unit at the end user's side. This computing unit runs a dedicated player that communicates with different display devices to make sure the video signals are properly received and synchronized on each of the display devices. All components required to achieve the goals of Pilot 1 are depicted in Figure 2.

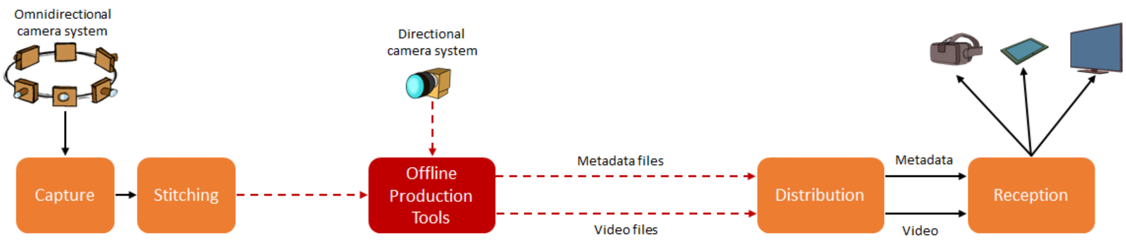

Figure 2: Architecture design of Pilot 1 (cf. D3.1)

This section describes the technical specifications of the different components in the architecture design and sums up the most noticeable issues that were observed during Pilot 1 (see table 1). The next section lists important lessons learned from these technical issues.

Design Components	Technical issues
1. Video Capture & Visualization	Some optical distortions in the 360 video material may be resolved with a better calibration and smarter camera setup.
2. Video Stitching	Some stitching errors would have been less visible if they had appeared in less sensitive parts of the scene (e.g. objects in the foreground or through smooth curves).
3. Post Production	Slow Adobe premiere plugin makes video editing more cumbersome.
4. Compression	Compression ratio is too high: distortion artifacts of different types were clearly visible in foreground objects (blocking, blurring, ringing, etc.).
5. Distribution	High-resolution content could not yet be handled by the DASH player.
6. Reception & Interaction	Synchronization between HMD, tablet and TV is sometimes disrupted when WiFi network is unstable.

Table 1: Technical issues per design component

3.1. Video capture

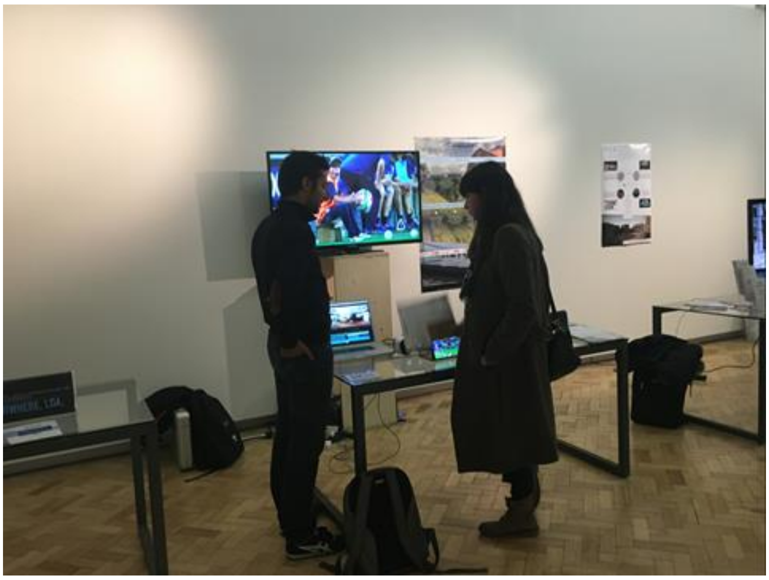

Figure 3: Display devices available to the end-user in Pilot 1

3.1.1. Camera specifications and setup

Pilot 1 deploys the one 6 Camera Rig with GoPro3 and one 3 Camera GoPro4 rig with a set of Entaniya 220 lenses[2] for off-line omnidirectional content capturing. These capture devices store several video streams on SD cards, which stitched and made available to off-line processing tools.

The H3PRO6 rig enables to combine 6 GoPro Hero 3 Black cameras together for capturing omnidirectional video streams. Each piece of the camera has a 12MPix CMOS sensor and produces H.264 encoded stream or provides HDMI live output. The cameras support storing on microSD/microSDHC cards in resolution up to 4K, although the frame rate in UHD resolutions is rather poor (12 or 15 fps). Each camera handles Full HD resolution in 60 fps (recording) or 30 fps (HDMI output). In this specific case, each camera recorded 2,7K at 23,98 fps. By using Pro Tune, we managed to balance the exposure between cameras.

The QBiC Panorama X camera rig enables to combine 4 QBiC cameras for capturing omnidirectional video streams. The camera is equipped with CMOS sensor and supports resolution up to Full HD in 60p. (recording) or 30 fps (HDMI output). The camera has WiFi output.

The most important parameters of the described camera systems with output capabilities are described and compared in Table 2 (cf. D3.1 - Architecture design).

3.1.2. Optical distortions

The spherical lenses in HMDs and in the capture devices induce optical distortions of which geometric distortions are the most common and important type. Geometric distortions in the 360-degree video material result in straight lines being perceived as skewed curves. Several scenes in the Pilot 1 demo suffered from clearly visible geometric distortions in certain angular

[2] https://www.entapano.com/en/l/panoramic_camera_panorama.html

perspectives (Figure 4) on both the tablet and the HMD. Some distortions at the capture side could have been avoided by improving the calibration quality of the cameras.

Camera Rig	# sensors	Output resolution
 **H3PRO6 Rig**	6	**Max resolution/frame rate** 6x 1920x1080p/60 fps when recording on SD card 6x 1920x1080p/30 fps on HDMI output
 **Elmo QBIC Panorama X rig**	4	**Max resolution/frame rate** 4x 1920x1080p/60 fps when recording on SD card 4x 1920x1080p/30 fps on HDMI output

Table 2: Pilot 1 camera specifications

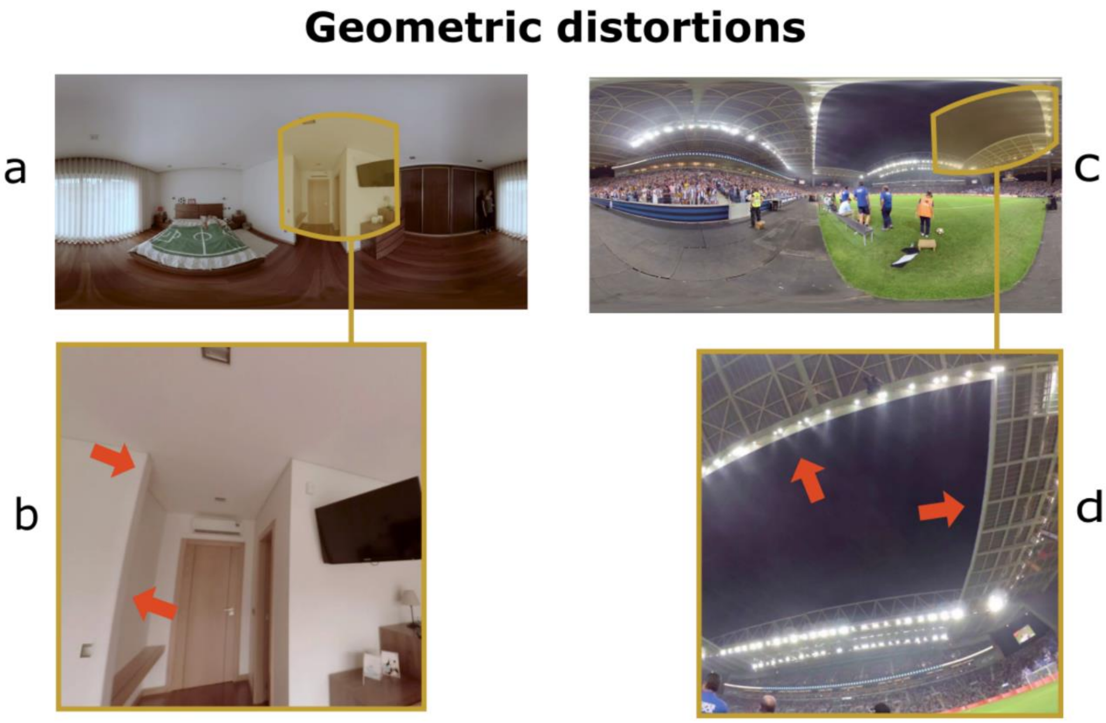

Figure 4: Geometric distortions observed in Pilot 1. (a) A video frame with equirectangular projection. (b) Zoom-in patch of the frame_(a) for better visualization of geometric distortion. (c) A video frame with equirectangular projection and (d) Zoom-in patch of the frame_(c) for better visualization of geometric distortion.

3.2. Video stitching

3.2.1. Stitching software

The original plan for Pilot 1 was to use VideoStitch to merge the footages of all cameras in the rig to a 360-degree video. However, Videostitch studio software was only available for Windows and could not be integrated into the MacOS-based pre and post-production framework of Lightbox. As a result, Lightbox resorted to a software from outside the ImmersiaTV consortium, named Kolor Autopano (www.kolor.com/autopano/).

3.2.2. Stitching errors

While stitching errors occurred with approximately the same frequency in every scene of pilot 1, they were far more visible in sensitive areas such as foreground objects and complex edge patterns. Since not all scenes are as easy to stitch and render, there should be given clear guidelines from the post-production side to the pre-production side about the scene composition, camera calibration, camera position, etc. For example, the camera operator should be aware that the parallax effect influences the stitching performance and can create artifacts (ghosting). Parallax is defined as the difference in alignment between two objects, when observed from different viewpoints. The parallax effect is happening when the images are not taken from the same place. The parallax can introduce stitching errors (by making seams more visible), especially when the objects are close to camera.

In case of static scenes, it is important to think about the location of camera to get less stitching errors. For example, in Figure 5, the stitching errors are visible in the pattern of the goal net. Moreover, this part of the scene has a large depth of field (the net, the ball and the background are in different depths), which complicates the stitching process even more. The stitching errors would have been less visible if the cameras were reoriented so that grass patches mainly needed to be stitched.

Another solution is to protect certain areas in a scene using a mask, such as the face of the football player in Figure 5b.

Unfortunately, relocating the camera rig or applying masks may not solve stitching problems when an object is moving or the scene is dynamic due to camera movements. This is, for example, the case for the crisp bridge discontinuity in Figure 5d. However, as the bridge is a part of the background, the stitching error could have been made less visible by applying a local blurring filter or another blending technique.

Besides a smoother blending of the different segments of the scene, one should always check that the colour information of the stitched fragments match. In Figure 5c, there is an abrupt change of colour saturation which was caused by different exposures of the cameras. The differences in exposures should have been compensated for. The VideoStitch Studio[3] software enables automatic exposure compensation/correction and it can be a solution.

[3] http://www.video-stitch.com/studio/

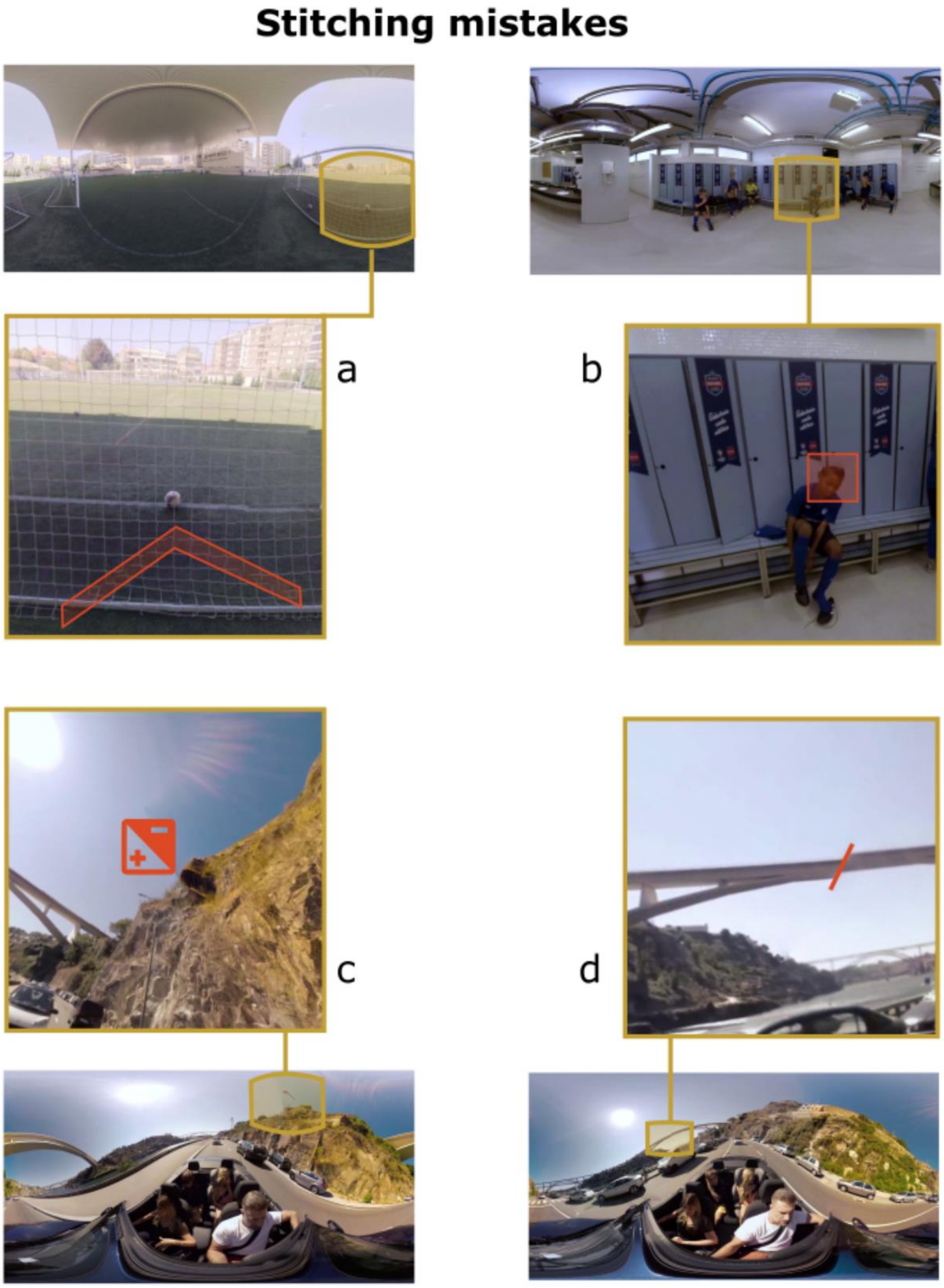

Figure 5: Stitching mistakes. Video frames and zoom-in patches for better visualization of the stitching errors. (a) Error in football net. (b) Error in person face. (c) Colour mismatch of stitched fragments. (d) Bridge discontinuity.

3.3. Video editing

3.3.1. Workflow

After video capture and stitching, offline video editing is performed. The video editing consists of a set of tools and plugins with the functionality of synchronization of multiple 2D videos, omnidirectional videos, and auxiliary data. These data come from the Video Acquisition, and Stitching blocks. Video editing, in general, is a complex process with many stages. In the project, typical media creation is extended by adding new technical possibilities, however they also impose some restrictions on the content creation process. All of the required additional functionalities are implemented and added to Adobe Premiere Pro as a set of plugins.

There are three main stages added to the standard editing workflow:

- synchronization of media for different output destinations
- defining portals/transitions/interactive points
- exporting to different output formats

By adding three stages to the typical content edition workflow, separate processes for each device (TV, tablet, HMD) are merged into one bigger process.

3.3.2. User friendliness of the Adobe Premiere plugin

The user friendliness issues of the plugin can be summarized as follows:

- Plugin Adobe Premiere is slow which makes rendering slow for post-production

Plug-in drains too much power from the machines. Small edits take huge time.

- Design of scene transitions not yet intuitive enough

The Adobe Premiere plug-in has improved significantly from its inception. At first, the plug-in only allowed to add static portals on an omnidirectional content. These portals offered the possibility of combining and synchronizing 2D contents with 360 videos. Subsequently, interactivity functionality was added to the portals offering, for instance, the possibility of making a transition from one scene to another or making a portal appear or disappear.

In addition to the features discussed above, currently, this plugin offers the possibility of adding input and output transitions to the different scenes, making the effect much friendlier.

Finally, this plugin allows the content creator to select what content they want to export from the project and for what devices they want to be available, therefore speeding up the edition of multi-platform content.

A preview of an edited scene can be observed in the Program Monitor window. Selecting the Portal effect in the Effect Control window enables overlay in the preview that visualizes parameters of a portal and allows their direct modification.

3.4. Compression

Compression is required in order to allow efficient streaming of the content after stitching and video editing. In the first iteration (offline encoding with minimal restrictions), ImmersiaTV adopted H.264 as the choice codec due, in part, to the ubiquitous availability of compatible decoders.

In pilot 1, many compression artifacts were noticeable in the 360 videos played on both HMD and Tablet. These artifacts include blocking, colour artifacts, ringing, and blurring.

Blocking artifacts are rectangular blocks in highly compressed video material. They occur mostly in fast motion sequences or quick scene changes. Figure 6a shows clearly visible blocking artifacts in the classroom scene of pilot 1. The artifacts on the right side are marked in red using an edge detector.

Compression artifacts in the chroma color channels lead to distorted colours. In the metro station scene, the footballer's arm appears to be partly green (Figure 6b).

Ringing artifacts are distortions near sharp transitions in a signal. They are caused by a loss of high-frequency information. Visually, they appear as bands or "ghosts" near edges. The breakfast scene in Figure 6c contains many ringing artifacts, which are marked in red on the right side using an edge detector.

The high compression ratio also causes sharpness defects and loss of details. The foreground objects in Figure 6d are blurred and some details only become more visible after sharpening, as shown on the right side.

The decrease in resolution leads to a loss of important details in the video. One of the problems is that the equirectangular projection magnifies less important parts of the scene such as the floor and ceiling and reduces the size of objects in the center of the scene. By reducing the resolution in such a projection, many foreground details will be removed. A region-aware remapping of the equirectangular projected image before encoding can mitigate this problem. To improve the quality, a remapping of the equirectangular projection should put more emphasis on important parts of the scene.

3.5. Distribution

In Pilot 1, due to Dash server limitations, it was necessary to use limited resolution for the content delivered to HMD and Tablet.

The IBC Congress (September 2016 Amsterdam) demonstration contained a 360 video (1080p, 8Kbps), TV scene (1080p, 3Kbps) and portals (640x360, 1.5Kbps). The videos are encoded at 25 FPS (frames per second), 25 GOP (Group of pictures). The segment length of the MPEG-DASH content is 3 seconds.

The next demonstration was shown in the NEM Summit (Nov. 2016 Porto). The demo contained 360 video (1080p, 3.7Mbps), TV scene (1080p, 2.7Mbps) and only one portal (340x180, 4Kbps) in the tablet and the HMD (Head-mounted display). The contents were encoded using the same parameters used at the IBC congress.

The latest, more recent demonstration setups tested in the laboratory aimed to improve the quality of the videos. We have already run a 360 video (2560x1440, 4Mbps), TV scene (1080p, 2.7Mbps) and only one portal (640x360, 6Kbps) in the tablet and the HMD (Head-mounted display). The technical test videos are encoded using the same parameters used at the IBC congress. The recent test setup delivers a better experience because the spherical videos are UHD. These latest technical tests have not been evaluated with end-users.

In Figure 7a the faces of the main actors in the scene are not well recognizable and details are lost. In Figure 7b the texts on the board in the center of the class are not readable.

To improve the end-user experience, the resolution should be increased in the next pilot.

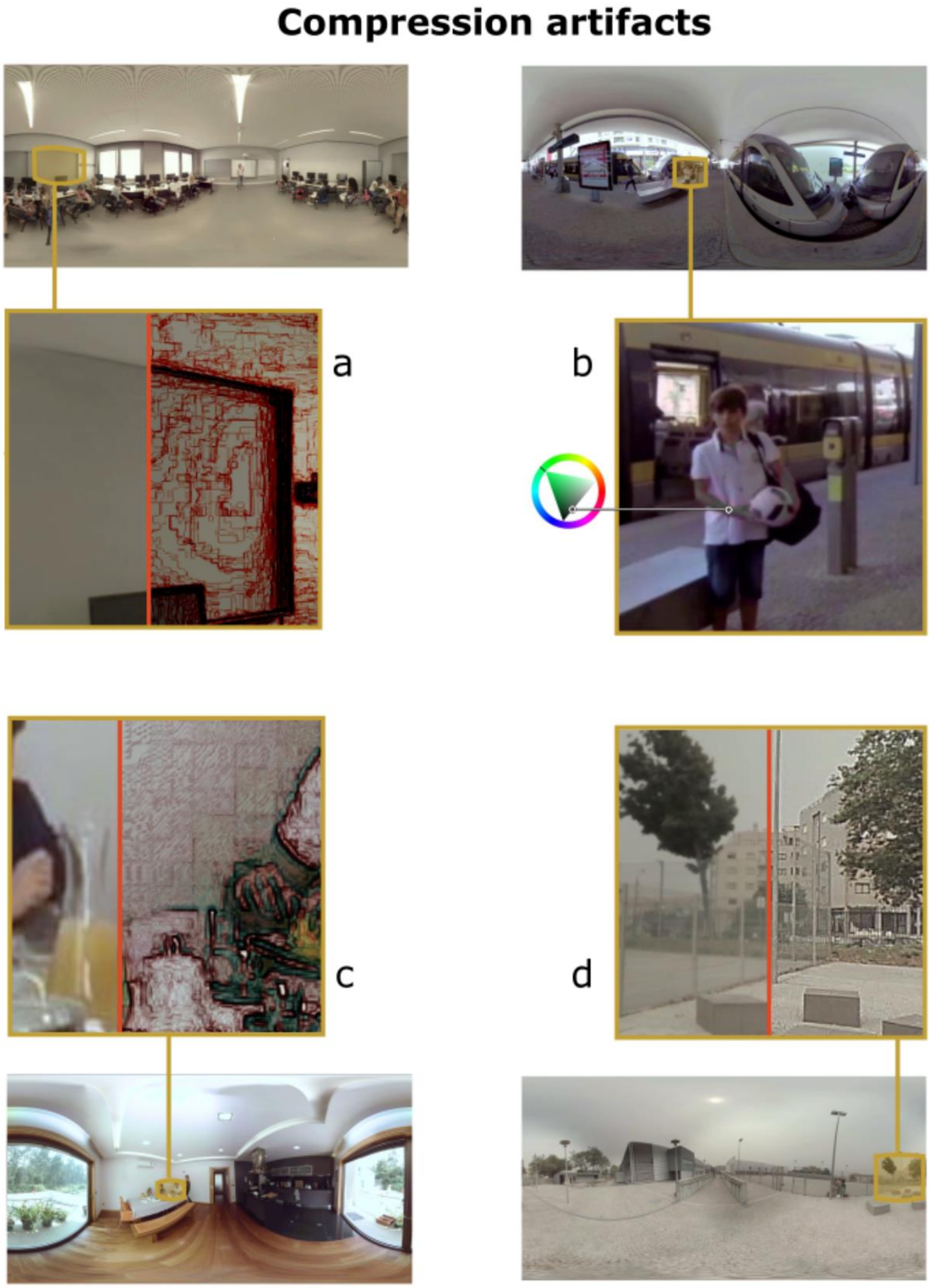

Figure 6: Compression artifacts: (a) A video frame and zoom-in patch for better visualization of the blocking artifact. (b) A video frame and zoom-in patch for better visualization of the colour distortion. (c) A video frame and zoom-in patch for better visualization of the ringing artifact. (d) A video frame and zoom-in patch for better visualization of the blurring artifact.

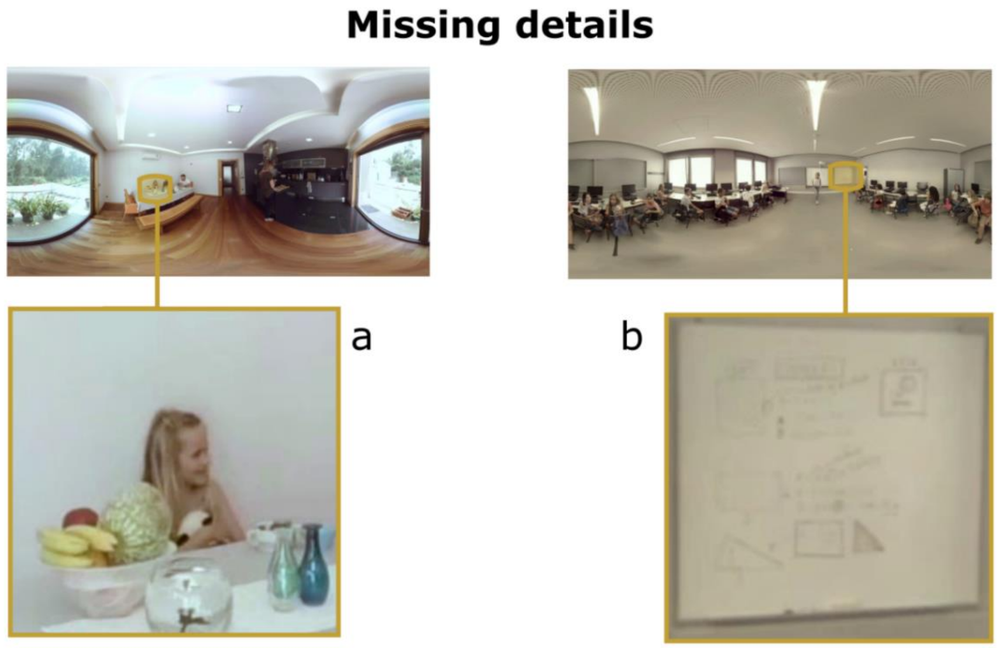

Figure 7: Missing details: The face details in (a) and. the texts in (b) are missing.

Tests: different encoding settings, different frame rate

The table 3 shows the characteristics of the videos that have been used to perform the tests.

	360 Video			TV			Portal		
Test vectors	Resolution	FPS	Bitrate (Mbps)	Resolution	FPS	Bitrate (Mbps)	Resolution	FPS	Bitrate (Mbps)
1SP_1080p_TV_1080_1PT_360p	1024x512	25	2.3	1024x512	25	2.9	426x240	25	0.638
1SP_2K_TV_2k_1PT_360p	2560x1440	25	3.7	2560x1440	25	3.5	426x240	25	0.638
1SP_4K_TV_4k_1PT_360p	3840x2160	25	3.7	3840x2160	25	4.3	426x240	25	0.638

Table 3: The characteristics of the test videos

Below we present the table containing the measurements of the images per second. Tests have been performed for different devices and different playback modes (tablet, TV or HMD). Note that the specified content is shown in the above table.

	SamsungGalaxy S7 (Tablet mode)	SamsungGalaxy S7 (HMD mode)	SamsungGalaxy S6 (Tablet mode)	SamsungGalaxy S6 (HMD mode)
1SP_4K_TV_4k_1PT_360p	29.9-33.1	18.2-24.7	15.2-20.0	12.1-13.5
1SP_2K_TV_2k_1PT_360p	38.9-57.9	27.3-33.4	25.4-31.2	16.5-20.7
1SP_1080p_TV_1080_1PT_360p	56.7-61.8	35.6-45.7	26.6-35.1	22.2-30.8
	Nexus 7 (Tablet mode)	Nexus 7 (TV mode)	SamsungGalaxy S6 (Tablet mode)	SamsungGalaxy S6 (HMD mode)
1SP_4K_TV_4k_1PT_360p	Not supported	Not supported	Not supported	Not supported
1SP_2K_TV_2k_1PT_360p	Not supported	Not supported	7.9-10.2	12.1-16.8
1SP_1080p_TV_1080_1PT_360p	9.0-12.2	10.1-13.5	14.4-17.7	24.0-29.3
	Android TV	Pixel C (Tablet mode)	Pixel C (TV mode)	
1SP_4K_TV_4k_1PT_360p	7.1-10.6	3.2-4.2	3.9-4.6	
1SP_2K_TV_2k_1PT_360p	24.5-28.7	7.2-8.7	8.8-9.6	
1SP_1080p_TV_1080_1PT_360p	44.8-48.3	8.9-12.0	12.1-15.2	

Table 4: measurements per second

3.6. Reception and interaction

In the on-demand scenario of pilot 1, videos were published on HTTP server. After connecting to the server the player presented a list of available content. Each content included versions of video dedicated for different devices – directive video for TV, omnidirectional videos for tablet and HMD. Users could start playout by selecting one position from the list. All the connected devices (TV, tablet, HMD) were presenting the same selected content (the same story), but each of them in a version dedicated for that device. On HMD users could freely select a direction to look at by movement of the head. On tablets they could do this moving tablet around or on the touch screen.

The main concern for pilot 1 was synchronization between HMD, TV and Tablet. In the unstable network environment, that we encountered during the demonstration in IBC (Amsterdam), the streams sent to HMD and tablet were delayed and easily got out of sync with the TV. The synchronization problem could not be further investigated when it occurred and the temporary solution was to reduce the bitrate of the transmitted signal and number of receivers. Later tests and developments have helped to improve this aspect, resulting in proper synchronization during the pilot in NEM Summit (Porto). For pilot 1 initial synchronization between HMD, TV and Tablet took several seconds, but as soon as it was achieved no further synchronization issues were observed. It is now possible to synchronize much faster – in less than a second.

Solving synchronization issues allowed focusing on increasing resolution of content playout. It was also a key point for presenting multiple video streams on the same device e.g. additional directive view composed into omnidirectional view. Lack of synchronization between videos would be even more disturbing than on separate devices. During pilot 1 it was presented in a limited way - only one video insert at a time was visible on Tablet and HMD. The interaction was limited to switching video insert on and off.

For the next Pilot, we need a more durable solution based on adaptively changing the bitrate of the signal based on the quality of network connection using the DASH protocol. Currently, the DASH player is not yet configured to switch between different quality presets.

4. LESSONS LEARNED FROM PILOT 1

4.1. Guidelines

According to the observed technical issues in Pilot 1, a number of guidelines are provided to improve performance of different stages of ImmersiaTV platform.

4.1.1. Shooting

The guidelines for shooting are summarized as follows:

Never shoot omnidirectional and directional scenes meant to be synchronized separately.

Shooting the scenes independently is problematic (for instance, first with the 360 Rig and next with the directional cameras). The problem is that adequate synchronization will never be possible because the acting (movements, moving objects, timings,...etc) will never be exactly the same between takes. As well, it is important to use same frame rate for all cameras during the capturing process.

The best way to shoot a documentary with omnidirectional and directional cameras simultaneously would be to have some micro 2k cameras around the set, hidden by props in the set design. This approach still necessitates some rotoscoping of the micro cameras to remove them from shot but this will be easier than removing an entire crew with filming equipment.

Put the directional cameras and crew in an area where there will be no interaction.

Another good approach is to put the directional cameras and crew in an area where there will be no interaction of characters or moving objects. In this case, another take of the same scene without the crew and directional cameras suffices to be able to remove them in post-production.

Pick up the tripod holding the 360 rig at the beginning of each take and rotate it 180 to 360 degrees, making regular pauses.

A good method to apply on the shooting set is to pick up the tripod holding the 360 rig at the end of each take and rotate it in 180 to 360 degrees, making regular pauses. Later in post-production, this movement of the cameras on the rig will give a clear idea if the cameras are synchronized correctly (by aligning the motion of all the captured videos). In addition, the pauses between rotations will image the same scene by each camera sensor, and will give more opportunities to correctly calibrate the rig.

Use a live preview system to review the scene composition before shooting.

It is recommended to use a quick or live preview system, such as an inexpensive camera like the Ricoh Theta S, or a better rig connected to Vahana VR, to let the director have a preview of the scene composition and the placement of actors: spherical video has no concept of “framing” or “zooming on” a subject since the entire sphere is covered at once. The director no longer has this liberty to guide the users’ attention. If stitching process in a post-production step does not made on location, and the composition of the scene is not suit the director’s intent, the take has to be repeated, however, this is not always possible, due to time and resource constraints.

4.1.2. Video calibration and stitching

- **Carefully pick up a calibration scene:** For automatic calibration (using VideoStitch Studio or Vahana VR for instance), it is better to choose the footage sequence with suitable content. This means: Choose a sequence that includes enough details (if a camera is shooting 100% the sky, it will be hard to calibrate). Moreover, check a sequence where there are no close objects in the video.
- **Calibration on several frames:** VideoStitch Studio enables calibration on several frames, if the calibration on a sequence failed.
- **Rotate your rig and launch the calibration again:** By rotating the camera, new details can be obtained on the overlapping part of the images, which will enable VideoStitch Studio or Vahana VR to detect new control points. The newly created control points will accumulate with the previous ones providing a better chance to get a successful calibration.
- **Using an external calibration software such as PTGui:** If the automatic calibration is still producing unsatisfactory results, one solution is to export individual frames from the input video and use an external calibration software, where users can interactively select and review control points. PTGUI is a helpful software tool to make a more perfect stitch of the camera rig output. Especially for parallax issues between cameras.
- **Move camera rig at the beginning of the take:** By moving the camera rig at the end of the take, the VideoStitch software gives a more accurate result on the synchronization, by aligning the motion of all the captured video. By changing the position of each video on the timeline the synchronization can be further optimized before starting the stitching process. This way, and finally, we will have the perfect synchronization to finally move to the stitching.
- **Keep the actors at least three meters from the camera rig:** Stitching solutions are still in their infancy, e.g. minor parallax problems are difficult to avoid for now. Keeping the actors at least three meters from the camera rig can leave these stitching and parallax problems largely unnoticed, by making a perfect stitch for the actors and ending up with some bad stitching in unimportant areas with no character interaction in the scene. Minor parallax stitching issues can be resolved in AfterEffects and Photoshop. Such issues can be corrected by painting out what we don't want to be seen and fixing the parallax problems.
- Stitching issues are more severe on certain locations (such as faces, patterns, etc), so it is more efficient to protect such areas by performing a conscious stitching.

4.1.3. Post-production

A good communication between preproduction and postproduction sides is needed because preproduction setups (camera adjustments, scene details and object distance from the lens, etc) can influence the complexity of postproduction.

4.1.4. Video compression, transmission, and reception

- Adaptive streaming can help to mitigate synchronization issues in case of an unstable network.
- A conscious remapping of the equirectangular projection needs to be done to avoid missing details. Some regions of the scene contains visually important details (for example the face details of an actor or texts, etc), while they occupy a very small portion

of the scene in an equirectangular projection. Therefore, it is important to detect such areas and perform an adaptive projection to preserve such areas from sever distortion.

- Higher resolution is needed particularly in omnidirectional videos to improve visual quality. Frame rate increase from 25Hz to 50Hz is recommended on HMD and Tablets.

4.2. Desirable features

In addition to the aforementioned guidelines and suggestions, some new features can be included in the ImmersiaTV platform to increase the end-user satisfaction:

4.2.1. 3D immersive audio

The 3D immersive audio recording is meant to reflect the way we receive sound in real life, creating rich soundscapes you would experience if you were actually there.

In past decades, 3D audio recording was a novelty which was utilized for less technically demanding methods. But with the rise of virtual reality hardware like the Oculus Rift, Sony's Morpheus, and Samsung's Gear — systems dependent on realistic 3D audio to fully immerse their users — 3D audio can have significant influence to increase the immersive experience. 3D immersive audio is becoming an important tool in virtual reality development.

4.2.2. Stereoscopic content

By generating stereoscopic content, the illusion of depth is created which helps for better immersive experience. However, there may be some challenges to use Stereoscopic 3D in VR. An important issue is to have a proper geometric arrangement of stereo cameras. When moving camera rig, the left and right camera must be horizontal to each other. Any small stitching misalignment (error) magnifies in 3D which leads to visual discomfort for end users. Inaccurate implemented 3D immersive video footage can cause a great discomfort to the viewer including eye strain or nausea.

4.2.3. High Dynamic Range

High Dynamic Range Imaging (HDRI) has recently gained attention and is affecting almost all fields of digital imaging. The HDRI overcomes the limitation of traditional imaging by performing operations on color data with much higher precision. The HDRI can represent all colors of real world close to what can be perceived by the human eye. Today many state-of-the art video game engines perform rendering using HDR precision to provide more believable virtual reality imagery.

5. STATUS OF USER REQUIREMENTS IN PILOT1

In the Deliverable D 2.3[4] - Content ideation, production scenarios and requirement analysis, four general user scenarios (US) of pilot1 with their specific requirements have been identified. This section summarizes the status of the requirements mentioned in D 2.3 (section 4).

US1. Story Preparation

Requirement Description	Current Status and Future Steps
R-STORY-1 The content creator can create the main storyline.	It is possible to create the main storyline. However, this new narrative paradigm requires experimentation and reflection in order to improve this particular point.
R-STORY-2 The content creator can define the main and side characters.	This feature depends mainly on the last point and how one makes use of Immersia's unique features.
R-STORY-3 The content creator can define the detailed story structure.	Due to the lack of information and knowledge in terms of narrative creation with this specific medium, this is still up for debate. Further experimentation will show how detailed one can craft a story within these constructs.
R-STORY-4 The content creator can define the sub-storylines for TV, tablet and HMD.	At this moment, this features shows the highest potential. Crafting nonlinear and time continuum disrupting narratives may be the key to help us understand this.
R-STORY-5 The content creator can define the user interaction design.	In terms of the plugin, this is still being worked. Hopefully in the near future we will be able to test it with test audiences to better understand how to craft a better interaction.
R-STORY-6 The content creator can define the multi-platform logic.	It is possible to define the multi-platform logic, when it is correctly implemented with the story.
R-STORY-7 The content creator can define the viewer perspective(s).	It is possible to define the viewer perspective(s). One of the best working aspects at the moment.
R-STORY-8 The content creator can define the detailed script.	The script must contemplate all aspects beforehand, otherwise, the risk of lack of narrative material is substantial. This might create a somewhat "dry" experience.
R-STORY-9 The content creator can define if it uses omnidirectional and/or directive content.	The feature is working perfectly at the present moment. Some technological improvement is still required.

[4] http://www.immersiatv.eu/wp-content/uploads/2015/11/D2.3-Content-ideation-production-scenarios -version-0.9.pdf

R-STORY-10 The content creator can define the viewing angle.	The feature is working well. Good exercise on points of view for any given narrative.
R-STORY-11 The content creator can define the interaction points: portal, AR object, caption, graphics, …etc.	Currently, the main issue is how counter-intuitive the organization of this material has to be conducted within premiere pro. Looking forward to, through collaboration, getting this point as optimized as possible.
R-STORY-12 The content creator can define transition between scenes.	At the moment, we can't give exact feedback whether this feature is correctly working or not.
R-STORY-13 The content creator can specify use of audio for guidance and transitions.	The feature is working correctly. It plays a very important role. However, it's currently not viable.
R-STORY-14 The content creator can indicate use of camera movements.	Presently, camera movement on the omnidirectional is still something we doubt it works correctly. Further testing will be conducted during Pilot 3.
R-STORY-15 The content creator can indicate forced exploration mode where applicable.	Not available, it is in the scope of pilot 3.
R-STORY-16 The content creator can save resulting format script as master template.	Not available.

Table 5: Status of the requirements for story preparation (US1)

US2. Production Preparation

Requirement Description	Current Status and Future Steps
R-PREPROD-1 The content creator can add on-location research material as placeholder in format script.	The feature is possible now.
R-PREPROD-2 The content creator can perform first VR preview of relevant scenes.	It is possible to perform first VR preview. It allows the content creator to understand if the location is viable for 360º shooting.
R-PREPROD-3 The content creator can define the shooting plan.	It depends on the ability/possibility of articulating how to tell a story with those three platforms.
R-PREPROD-4 The content creator can define the VR/directive capturing strategy.	The content creator can define the VR/directive capturing strategy. However, some field testing has yet to be done to fully understand the extent of such strategies and their specific applications.

Table 6: Status of the requirements for production preparation (US2)

US3. Edition and Compositing

Requirement Description	Current Status and Future Steps
R-EDIT-1 The content creator can visualize the raw material across the different end-user devices.	The raw material can be visualized across end-user devices.
R-EDIT-2 The content creator can use a standard edition software (Adobe Premiere, Final Cut, or other), and avoid, for simple projects, using advance compositing software.	Using Adobe Premiere is enough for standard edition.
R-EDIT-3 In the editor software, the content creator can edit content for TV and for omnidirectional video in such a way that the timings of the content for the 2 targeted devices is visible constantly	TV and omnidirectional videos are separate tracks on a common timeline.
R-EDIT-4 The content creator can use Windows and OS X	The software versions for Windows and OS X are available.
R-EDIT-5 The content creator can use of an advanced mode in a compositing software (Nuke, Adobe After Effects).	Advanced mode in the editing software is not addressed directly, but it does not prevent user from enhancing content with tools other than Adobe Premiere.
R-EDIT-6 The content creator can introduce interactivity within the editor timeline through conditional transitions between shots and scenes.	The portal effect plugin allows interactivity.
R-EDIT-7 The content creator can select, within the editor timeline, which video assets are visible within the TV, the tablet and the HMD.	Creator selects the target device for video assets, by naming the video tracks.
R-EDIT-8 The content creator can also create ImmersiaTV scene typologies, i.e., interaction between devices, through conditional transitions within the editor timeline.	There are no interaction between devices currently, different scene typologies can be created.
R-EDIT-9 In pilot 1, the end user will experience the content with a common timing between devices (HMD, TV, tablet), it will be continuous and have no jumps.	The content for pilot 1 is continuous, with common timing for all devices.
R-EDIT-10 The content editor, using either a classic video editor or an advanced one, will easily define transitions between omnidirectional videos using black and white video MATTE.	Transitions are implemented as Adobe Premiere plugins with corresponding shaders on player side. Defining transition as black and white video matte is not available.
R-EDIT-11 The content editor will be able to add a beauty layer to the interactive transition which, unfolding synchronously with the black and white video matte, will add borders and eventually other visual content needed for the transition.	Not available.

R-EDIT-12 The content creator will allow seeing omnidirectional content both in projected and non-projected views by using Previsualisation tools integrated in the content editor.	Previewing in projected and non-projected mode is a built-in feature of latest Adobe Premiere Pro.
R-EDIT-13 The content creator will be able to visualize transitions and interactive transitions will be visible within the editing software.	Transitions are rendered in preview window.
R-EDIT-14 The content creator will be able to visualize synchronized playout between 2 devices, for example, to see how TV and HMD content fit in timing.	The feature is not addressed directly, but should be possible using built-in Adobe Premiere features.
R-EDIT-15 An export button will generate a set of videos and metadata that is ready to distribute content across devices.	Simple export panel allows to generate videos and metadata ready to be played in the player application.
R-EDIT-16 The export functionality will accept sequences involving different aspect ratios, due to differences in omnidirectional and traditional video formats (most likely solved through nested sequences).	The export functionality takes into account aspect ratios.
R-EDIT-17 The common cutting points between devices will be visualized putting the content for the different devices in 2 sequences, one on top of one another.	Content for different devices is edited in a common sequence, on different tracks, which allows to visualize common cutting points.
R-EDIT-18 It will be possible to define a label specifying the destination for each sequence.	Creator selects the target device for a sequence, by naming the video tracks.
R-EDIT-19 The outcome should be: 1) A set of videos in the highest resolution possible. The videos should have a shared timestamp. This means that the timestamp introduced at the frame level is common to all the different fluxes. For example, the first frame of a video introduced exactly at second 12 of the broadcast should have its first frame with a timestamp set at 12. 2) A metadata file detailing how the different videos have to be organised to compose an omnidirectional scene. This file should be compatible with broadband distribution standards.	The outcome of export is: - video files in highest possible resolution, with shared timestamp - metadata describing relations between videos, compatible with DASH

Table 7: Status of the requirements for editing and compositing (US3)

US4. Content Playback

Requirement Description	Current Status and Future Steps
R-PLAY-1 Basic controls. The basic controls of the player will be: Select media source, Play, Stop, Select tablet or HMD mode.	The media source can be selected from the player, current situation is that the user has to stop every device in order to playback a new content. The HMD or tablet behavior can be selected by going back to device selection screen.
R-PLAY-2 The player will process metadata to describe and define the scene: The information regarding how the scene is composed must be distributed to the player. It must include information like which videos are visible and where are they placed or how are they composed.	The player can process the metadata and is distributed from the same place the contents are originated. The metadata indicates the size, position, start time and duration of the scenes.
R-PLAY-3 The scene is device dependent. Each type of device will have to render a different scene, as the interaction with the user will be different.	There are three types of scenes per type of device (HMD, tablet and TV).
R-PLAY-4 Render multimedia content over textures and 3D objects. One or several videos will be displayed in different positions over the 3D scene (over a spherical surface, as a regular 360° video, or over plain surface in a mirror or portal like effect).	The videos are rendered in different parts of the scene, managed by Unity3D game engine.
R-PLAY-5 Apply video masks in videos. A mask is needed to overlay more than on video over the same texture forming an overlay of an arbitrary shape (i.e. to render a portal as a circle over the 360° sphere).	The masks are now rendered from luma matte videos. For next pilot they will be generated from predefined transitions and applied from the player.
R-PLAY-6 Interaction management. There needs to be a systematic way to define interaction mechanisms in the end-user devices, and the methods implementing such interaction mechanisms need to be made available to the content creator.	The interactions are defined in the Premiere Pro plugin with the portal effect plugin.
R-PLAY-7 Achieve a frame level precision: This is relevant as devices can display different omnidirectional and directional contents that were shot together, so any sort of desynchronization is going to be noticeable by the user.	Frame accurate synchronization has been achieved.
R-PLAY-8 The devices may need to synchronize to any base media time at start up: A device can be turned on when there is already the reproduction going on in another device, so the one joining the group must get synchronized without affecting the other ongoing reproductions.	When a device joins the common session it synchronizes to the playback point of the others (previously synchronized).
R-PLAY-9 Basic audio control in the end-user devices	Stereo audio is available and can be muted from the device.

R-PLAY-10 Real time communication channel between devices: It will be needed to send messages from one device to another.	There is no mechanism for inter-device messaging currently.
R-PLAY-11 Second screen scene definition: The definition of the second screen view (mosaic layout) in the tablet must be defined within the content production process.	The mosaic layout is not available yet, it will be implemented for pilot 2.
R-PLAY-12 The end-user can capture screen casts and share them with other devices.	Not available, it will be implemented for pilot 3.
R-PLAY-13 The end-user can capture screen casts and share them through social media.	Not available, it will be implemented for pilot 3.

Table 8: Status of the requirements for content playback (US4)

6. CONCLUSION

This document presented an overview of ImmersiaTV platform and components specifications, as well as the technical issues observed in Pilot 1. Moreover, a number of guidelines are provided to improve the pre-production and post-production performance.

Some important guidelines are summarized as follows: In shooting (video capture), the 360 rig and directional camera should operate simultaneously to avoid synchronization issues. It is required to carefully pick up a calibration scene. A number of recommendations are provided in the document for better calibration which can also lead to improved stitching performance. A good production plan is needed because preproduction setups (such as camera adjustments, and scene details) can influence the complexity of postproduction (stitching). In video compression, a content aware remapping can avoid missing detail. Higher resolution and frame rate are recommended to improve the quality of experience.