



AI기반 스타일 전이 기술을 활용한 360VR 콘텐츠 제작_2day

이하은 강사

MR미디어 아트텍센터 수석연구원
홍익대학교 영상커뮤니케이션대학원 VRAR콘텐츠전공



CONTENTS

01 1day Summary

- AI스타일전이/ VR의 이해/ 360Video vs. VR
- 실습 내용 요약 (360 사진,영상 전환 및 제어)
- Shader이용 뒤집기/영역에 따른 사진,영상 전환

02 팀별 계획서 작성

- 계획서 예시자료
- 팀별 제출 기획서 확인 및 촬영장소 선정
- 시나리오에 따른 워크플로우 및 개발일정 작성

03 360 Video

- 360 Camera
- 360 Video Processing

04 2일차 실습 내용

- 360 이미지 및 비디오 촬영
- 360 이미지 및 비디오 편집
- 영역에 따른 사진, 영상 전환 과제물 제출



01 1day Summary

커리큘럼 및 프로젝트 진행 일정

수업 계획서			
회차	수업주제 및 내용	수업방법	시수
1	1. 오리엔테이션 - VR의 이해 - AI(심층신경망) 기반 스타일전이 활용작품 소개 2. 스크립트 이용 360 사진 및 영상 전환 - 360 이미지 및 영상 출력 - 360 이미지 및 영상 전환	이론/실습	8
2	1. AI(심층신경망)기반 스타일전이의 이해 - 스타일전이를 위한 오브제영상과 스타일영상 선정 - 스타일전이 엔진 활용을 위한 영상소스 제작 2. 360영상 촬영실습 - 360 영상 및 이미지 촬영 - 360 이미지 편집 실습	이론/실습	8
3	1. 팀별 프로젝트 계획서 작성 - 예시를 통한 작성 방법 확인 - 계획서 확인 후 팀별 피드백 2. UI 및 3D Object와의 상호작용 - 컨트롤러 입력 및 핸드트래킹 - Lighting, Sound, Character 추가 기능	이론/실습	8
4	1. 팀별 프로젝트 발표준비 - 스타일전이 결과물 확인 - 팀별 프로젝트 제작 마무리 2. 팀별 프로젝트 발표 및 시연	이론/실습	8

촬영장소: DMC명소 5곳
총 인원(20명): 5개 팀 * 4명씩
팀당 360VR 1종 제출
팀당 360 이미지 촬영 1곳

일일 수업결과물

(10일) 포탈 활용 이미지 전환

(11일) 360촬영 결과 HMD확인

(12일) 프로젝트 계획서

(17일) 프로젝트 최종 결과물



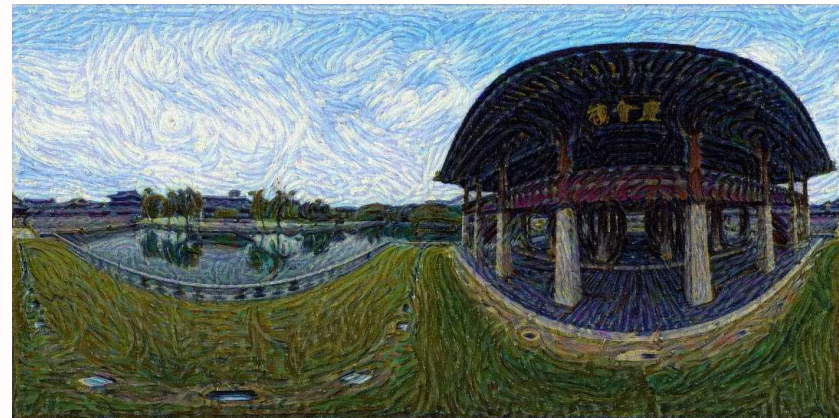
AI Style transfer

인공지능 스타일 전이는 오브제(Content) 영상과 스타일(Style)영상을 입력으로 받아 스타일 전이된 결과물 영상을 얻는 기술



범용 스타일 전이 엔진(딥아트 : depart.io)조차도 아직까지는 초고해상도 결과물 도출이 불가

초고화질 스타일전이엔진(stNET)의 결과물을 예상하기 위한 대략적인 느낌은 딥아트와 유사하니 계획서 작성 시 이를 반영하여 제출



01 1day Summary

VR의 유형(시스템 환경 별)



VR

(Virtual Reality)



몰입형
가상현실
(Immersion VR)



프로젝션형
가상현실
(Projection VR)



시뮬레이션형
가상현실
(Simulation VR)



데스크탑형
가상현실
(Desktop VR)



원거리
로보틱스



3인칭
가상현실

Triangle of Virtual Reality

• **Immersion :**
Feeling to be in the 3D Virtual Space.

• **Interaction :**
Possibility of moving in the 3D space and manipulate objects

• **Real Time :**
Actions can immediately modify the state of the space .

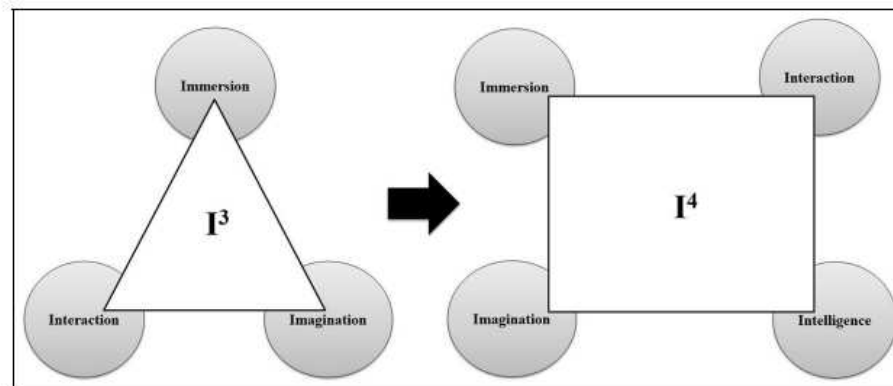
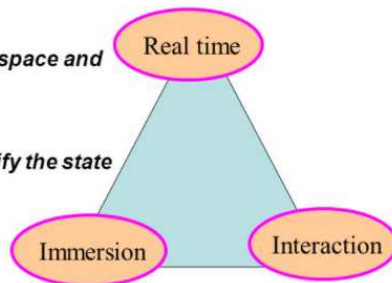


Figure 1. Virtual reality triangle transformation to virtual reality quadrilateral.

01 1day Summary

VR, AR, MR 그리고 XR 용어



MR (Mixed Reality)

가상세계와 현실세계를 합쳐서 새로운 환경이나 시각화 등 새로운 정보를 만들어 내는 것
본 개념은 VC(가상 연속체)에서의 미디어적 분류 개념으로 VR의 시공간에 한정적이라는 한계가 존재

RE (Real Environment)

현실 속 환경이나 상황

AR (Augmented Reality)

실제 환경에
가상 사물 및 정보를 합성해
원래 환경에 존재하는 것처럼 보이게 함

AV (Augmented Virtuality)

가상 환경에
실존하는 사물/인물을 합성해
원래 환경에 존재하는 것처럼 보이게 함

VE (Virtual Environment)

인공적 기술로 만들어 낸
실제와 유사하지만 실체가 아닌
어떤 특정한 환경이나 상황

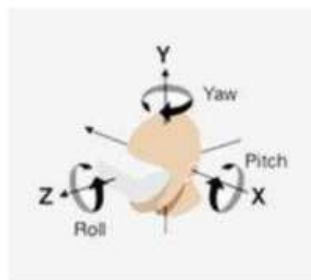
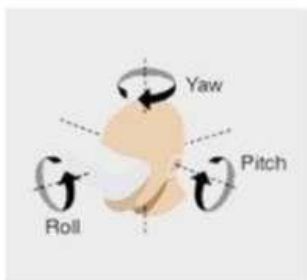
XR (eXtended Reality)

XR는 가상현실(VR), 증강현실(AR)을 결합한 혼합현실(MR)의 일종, 단순히 (HMD)을 머리에 쓰고 고개만 갸웃거리는 게 아니라,
XR는 손발과 온몸을 움직이면서도 현실세계와 가상세계를 넘나드는 것을 가능케 한다



3Dof vs 6Dof

3 dof	6 dof
Head orientation only	Head orientation + Walk around
360 Video	Full VR
Ambisonic sound	Object – based audio



Indoor vs. outdoor



Inside-out vs. Outside-in



Inside-Out Tracking

Outside-In Tracking

01 1day Summary

VR과 360Video의 비교



VIRTUAL REALITY vs. 360 VIDEO

PHOTOGRAPHY



Digital Environment

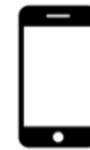


Live Action

PLATFORMS



Full experience requires a VR headset (can be "tethered" or mobile)



Available on 360 compatible players including YouTube (desktop and mobile)

MOBILITY



Immersive world that you can walk around in (as long as you are not "tethered" or connected to a computer)



360 degree view from camera's perspective, but limited to filmmaker's camera movements

STORY



Filmmaker does not control physical location of viewer in the built environment (as long as you are not tethered) and as such must capture attention and also motivate user to travel in the direction of the events of the story

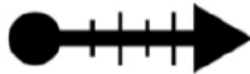


Filmmaker controls physical location of camera, but must capture attention of viewer to direct the story

VIDEO TIMELINE



Video can progress through a series of events or experience can be simply an existing world to be explored by the user



Video progresses on a timeline created by the filmmaker's camera movements



01 1day Summary

360 이미지 및 비디오 출력 실습_skybox/ sphere

360 이미지 출력_skybox

- 1) Project View에서 우클릭 후 [Create-Material] 선택
- 2) 생성된 Material의 Shader를 Skybox-Panoramic으로 변경
- 3) 설정한 Material의 Texture에 360이미지를 Drag&Drop
- 4) 만들어진 Material을 Scene에 Drag&Drop
(4번은 Window-Rendering-Lighting Settings 메뉴에서 Skybox Material에 만들어진 Material을 설정해도 됨)

360 비디오 출력_skybox

- 1) Project View에서 우클릭 후 [Create-Material] 선택
- 2) 생성된 Material의 Shader를 Skybox-Panoramic으로 변경
- 3) Project View에서 우클릭 후 [Create-Render Texture] 선택해 생성된 Render Texture의 Size에 import한 영상 크기 입력
- 4) 2번에서 설정한 Material의 Texture에 Render Texture를 Drag&Drop
- 5) Hierarchy뷰에서 우클릭 후 [Video-Video Player] 선택
- 6) 생성된 Video Player에서 Video Clip에 import된 360비디오를, Target Texture에 Render Texture를 Drag&Drop
- 7) 만들어진 Material을 Scene에 Drag&Drop
(4번은 Window-Rendering-Lighting Settings 메뉴에서 Skybox Material에 만들어진 Material을 설정해도 됨)



360 이미지 출력_sphere

- 1) Hierarchy View에서 우클릭 후 [3D Object-Sphere] 선택
- 2) Project View에서 우클릭 후 [Create-Material] 선택
- 3) 생성된 Material의 Shader를 Unlit-Texture로 변경
- 4) 설정한 Material의 Texture에 360이미지를 Drag&Drop
- 5) 변경된 Material을 만들어진 sphere에 Drag&Drop
- 6) ReverseNormals 스크립트 파일을 생성 후 이를 sphere에 컴포넌트로 추가

360 비디오 출력_sphere

- 1) Hierarchy View에서 우클릭 후 [3D Object-Sphere] 선택
- 2) Project View에서 우클릭 후 [Create-Material] 선택
- 3) 생성된 Material의 Shader를 Unlit-Texture로 변경
- 4) 변경된 Material을 만들어진 Sphere에 Drag&Drop
- 5) Sphere의 Inspector창에서 Add Component 클릭하여 Video Player를 추가하고 Video Clip엔 360비디오를 Drag&Drop
- 6) ReverseNormals 스크립트 파일을 생성 후 이를 sphere에 컴포넌트로 추가



01 1day Summary

360 비디오 제어 및 360 이미지 전환 실습(스크립트 활용)

** VideoPlayer 다루기

```
1 using UnityEngine;
2 using UnityEngine.Video;
3
4 // Unity 스크립트 참조 0개
5 public class VideoPlayerTest : MonoBehaviour
6 {
7     // 비디오 재생 목록
8     public VideoClip[] videoClip;
9     // VideoPlayer를 담을 변수
10    private VideoPlayer vp;
11    // 현재 재생 중인 videoClip index
12    private int currentClipIndex = 0;
13
14    // Unity 메시지 참조 0개
15    void Start()
16    {
17        // VideoPlayer 지정
18        vp = GetComponent<VideoPlayer>();
19        // 현재 재생 중인 비디오가 끝났을 때 알려주는 이벤트
20        vp.loopPointReached += OnMovieFinished;
21        // 처음 재생 될 비디오 설정
22        vp.clip = videoClip[currentClipIndex];
23        // 시작하자마자 Video를 재생시킬 것인지
24        vp.playOnAwake = false;
25        // 비디오를 반복 재생할 것인지
26        vp.isLooping = false;
27
28        vp.Pause();
29    }
30
31    // Unity 메시지 참조 0개
32    void Update()
33    {
34        Debug.Log(vp.targetCameraAlpha);
35
36        if (Input.GetKeyDown(KeyCode.Space))
37        {
38            // 비디오 일시정지, 재생
39            if (vp.isPlaying)
40                vp.Pause();
41            else
42                vp.Play();
43        }
44    }
45 }
```

```
41
42 else if (Input.GetKeyDown(KeyCode.A))
43 {
44     // 10초 스킵
45     vp.time += 10;
46 }
47 else if (Input.GetKeyDown(KeyCode.S))
48 {
49     // 10초 이전
50     vp.time -= 10;
51 }
52 else if (Input.GetKeyDown(KeyCode.D))
53 {
54     SetVideo();
55 }
56
57 // 재생 중인 비디오가 끝났을 때 실행
58 참조 1개
59 void OnMovieFinished(VideoPlayer vp)
60 {
61     SetVideo();
62 }
63
64 // 비디오 클립 바꾸는 함수
65 참조 2개
66 void SetVideo()
67 {
68     Debug.Log("Set Video");
69     vp.Stop();
70
71     if (currentClipIndex < videoClip.Length - 1)
72         currentClipIndex++;
73     else
74         currentClipIndex = 0;
75
76     vp.clip = videoClip[currentClipIndex];
77     vp.Play();
78 }
```

** 360 이미지 전환

```
1 using System.Collections;
2 using UnityEngine;
3
4 // Unity 스크립트 참조 0개
5 public class Image360 : MonoBehaviour
6 {
7     public OVRScreenFade fader;
8     public Material imageMat;
9
10    public Texture[] texture;
11    private int currentIndex = 0;
12
13    // Unity 메시지 참조 0개
14    void Start()
15    {
16        imageMat.mainTexture = texture[currentIndex];
17    }
18
19    // Unity 메시지 참조 0개
20    void Update()
21    {
22        if (Input.GetKeyDown(KeyCode.A))
23        {
24            StartCoroutine(ChangeTexture());
25        }
26    }
27
28    참조 1개
29    IEnumerator ChangeTexture()
30    {
31        if (currentIndex < texture.Length - 1)
32            currentIndex++;
33        else
34            currentIndex = 0;
35
36        yield return StartCoroutine(fader.Fade(0, 1));
37
38        imageMat.mainTexture = texture[currentIndex];
39
40        yield return StartCoroutine(fader.Fade(1, 0));
41    }
42 }
```



360 이미지 및 비디오 전환 실습_포탈 1개

- 1) Asset Store에서 Pocket Portal VR Asset 다운로드
- 2) Project뷰에 Empty Object를 만들어 World1로 이름 변경 후 Sphere(video)를 자식으로 위치
- 3) Project뷰에 Empty Object를 만들어 World2로 이름 변경 후 Sphere(Image)를 자식으로 위치
- 4) World1과 World2 오브젝트에 Dimention 컴포넌트를 검색해서 각각 추가
- 5) World1과 World2 중 먼저 보여주고 싶은 환경의 Dimension 내 Initial World를 체크
- 6) Project View에서 square-portal-vr 프리팹을 검색하여 Project뷰에 추가

** 360 이미지 및 비디오를 출력하는 두개의 구 위치는 정확히 일치해야 함



360 이미지 출력_sphere

- 1) Flip Normals Shader를 Project뷰에 Drag&Drop하여 Import
- 2) Hierarchy View에서 우클릭 후 [3D Object-Sphere] 선택
- 3) Project View에서 우클릭 후 [Create-Material] 선택
- 4) 생성된 Material의 Shader를 Custom-Flip Normals로 변경
- 5) 설정한 Material의 Texture에 360이미지를 Drag&Drop
- 6) 변경된 Material을 만들어진 sphere에 Drag&Drop
- 7) Window-Rendering-Lighting Settings 메뉴를 선택하여 우측 그림처럼 설정

Shader를 활용하면 flip하는 스크립트를 작성하지 않고도 클릭 두 번으로 같은 결과를 얻을 수 있음.

하지만, 이는 360Video를 감상할 때는 차이가 없어 보이지만 추후 360VR환경에서 3D오브젝트를 추가하면 Skybox의 lighting이 제외되어 sport light나 directional light를 비춰야 한다는 단점이 있음.





01 1day Summary

360 이미지 및 비디오 전환 실습(추가)_영역

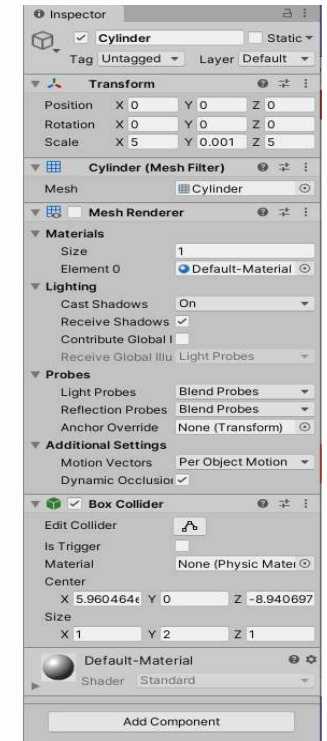
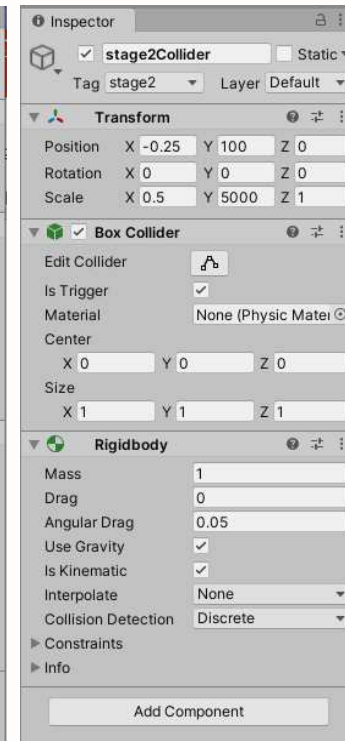
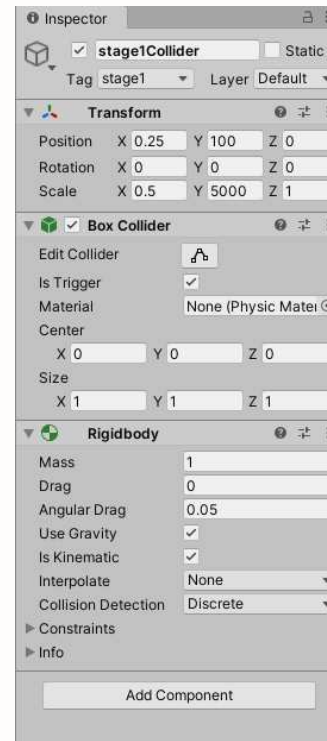
360 이미지 전환 실습_영역 2개

- 1) Cylinder 오브젝트 생성 후 Transform의 Scale을 5, 0.001, 5로 수정하고 Capsule Collider 컴포넌트를 삭제
- 2) Cylinder 오브젝트 자식으로 cube 2개를 생성하고 각각을 구분하기 위해 stage1Collider, stage2Collider로 이름변경
- 3) Inspector를 그림과 같이 컴포넌트 수정
- 4) OVRCameraRig>TrackingSpace>CenterEyeAnchor에 스크립트 작성 후 추가

```
StageCollCheck.cs
Assembly-CSharp

1 using System.Collections;
2 using System.Collections.Generic;
3 using UnityEngine;
4
5 public class StageCollCheck : MonoBehaviour
6 {
7     public Material material;
8     public Texture[] textures;
9
10    private int currentStage;
11    private bool isStart;
12
13    private void Start()
14    {
15        isStart = true;
16    }
17
18    //CameraRig 위치에 따른 360 Texture 변경
19    private void OnTriggerEnter(Collider other)
20    {
21        ChangeTexture(other);
22        Debug.Log("OnTriggerEnter");
23    }
24 }
```

```
//콘텐츠 실행 후 CameraRig 위치에 따른 360 Texture 초기화
25 private void OnTriggerStay(Collider other)
26 {
27     Debug.Log("OnTriggerStay");
28     if (!isStart)
29         return;
30
31     ChangeTexture(other);
32     isStart = false;
33 }
34
35 private void ChangeTexture(Collider space)
36 {
37     if (space.tag == "stage1")
38     {
39         currentStage = 0;
40     }
41     else if (space.tag == "stage2")
42     {
43         currentStage = 1;
44     }
45
46     material.mainTexture = textures[currentStage];
47 }
48
49
50 }
```



【 팀별 계획서 작성 】

02 팀별 계획서 작성

기획 내용에 따른 계획서 작성



주제: DMC 명소

전시 장소: DMC코워킹센터

* DMC명소 10곳의 360촬영결과물을 활용한 콘텐츠 전시 *

기획서

작품 명(제목), 팀원 명(역할),
작품소개: 기획의도, 콘텐츠 시나리오(사건 혹은 줄거리)

계획서

엔진 결과물 예상 이미지(오브제+스타일 그림 추가),
Flowchart(개발 범위/단계), 콘텐츠 개발 단계별 진행 계획(일정)

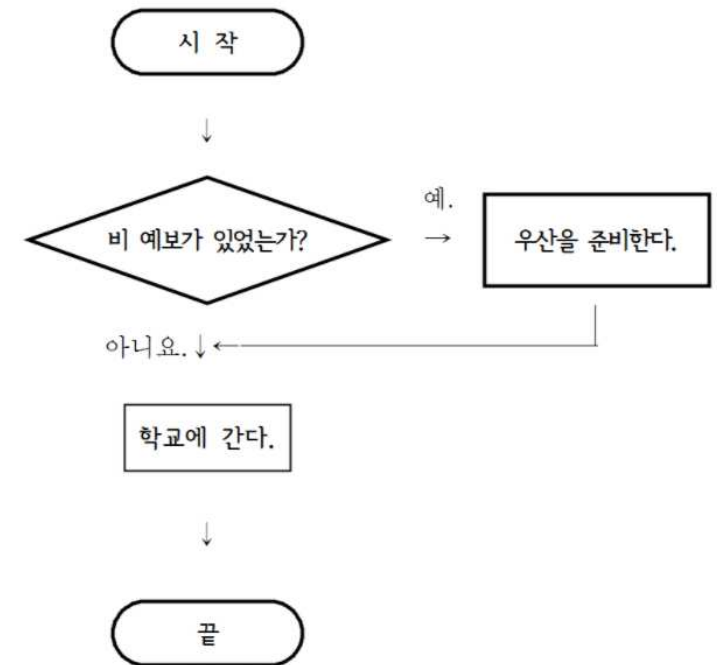
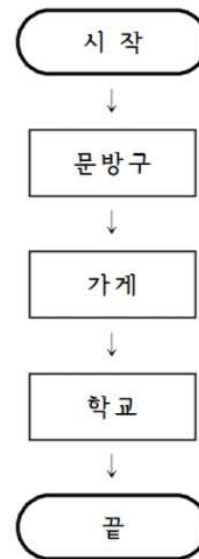
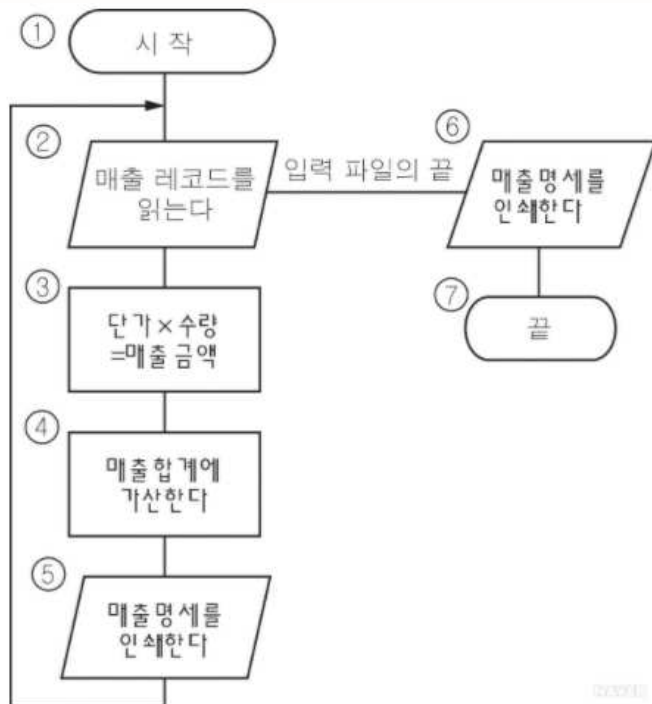
발표 및 시연

작품 발표 및 시연 시간: 팀당 8~10분(시연은 링크 활용 혹은 영상)
24일 : 5시~6시 작품 발표 후 직접 체험 진행



02 팀별 계획서 작성 기획 내용에 따른 계획서 작성

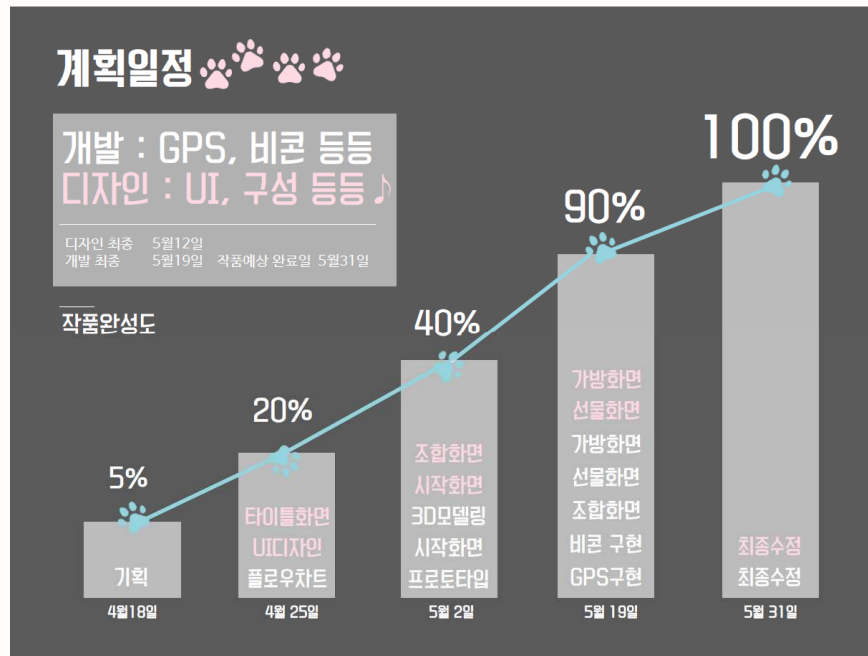
Flowchart



시나리오에 따른 워크플로우 및 개발일정 작성



02 팀별 계획서 작성 시나리오에 따른 워크플로우 및 개발 일정 작성



한계 · 보완 (체크 시스템) 🐾🐾🐾

DECEMBER 31

내가 맞춘 문제를 체크
개수, 타입별로 다른 보상

보상을 받기 위해 일정 개수 이상, 문제 타입별로 보상을
다르게 준비해 지속적인 플레이 유도

[360Video]



Types of 360 Camera

Single-Lens	Dual-Lens	Multiple Cameras
Mirrored Lens Fisheye Lens	Fisheye Lens	Rigged Embedded
Monoscopic(2D)	Stereoscopic(3D)	



Mirrored Lens



users can get a 360-degree horizontal perspective

The mirror and software takes something that fits on a small sensor and stretches it to create a 360-degree view — that's going to mean a loss in resolution and overall quality

03 360Video

360 Video Camera(~2016)



GoPro



**Richo
Theta 360**



Gear 360

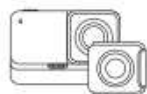


**Codac
SP 360**

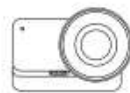


보급형

액션카메라



ONE R
트윈 에디션



ONE R
1인치 에디션
CO-ENGINEERED WITH 



GO



ONE X



ONE

360 / 3D 카메라



EVO



Nano S
For iPhone



Nano
For iPhone



Air
For Android

프로형

VR 카메라 프로



Pro 2



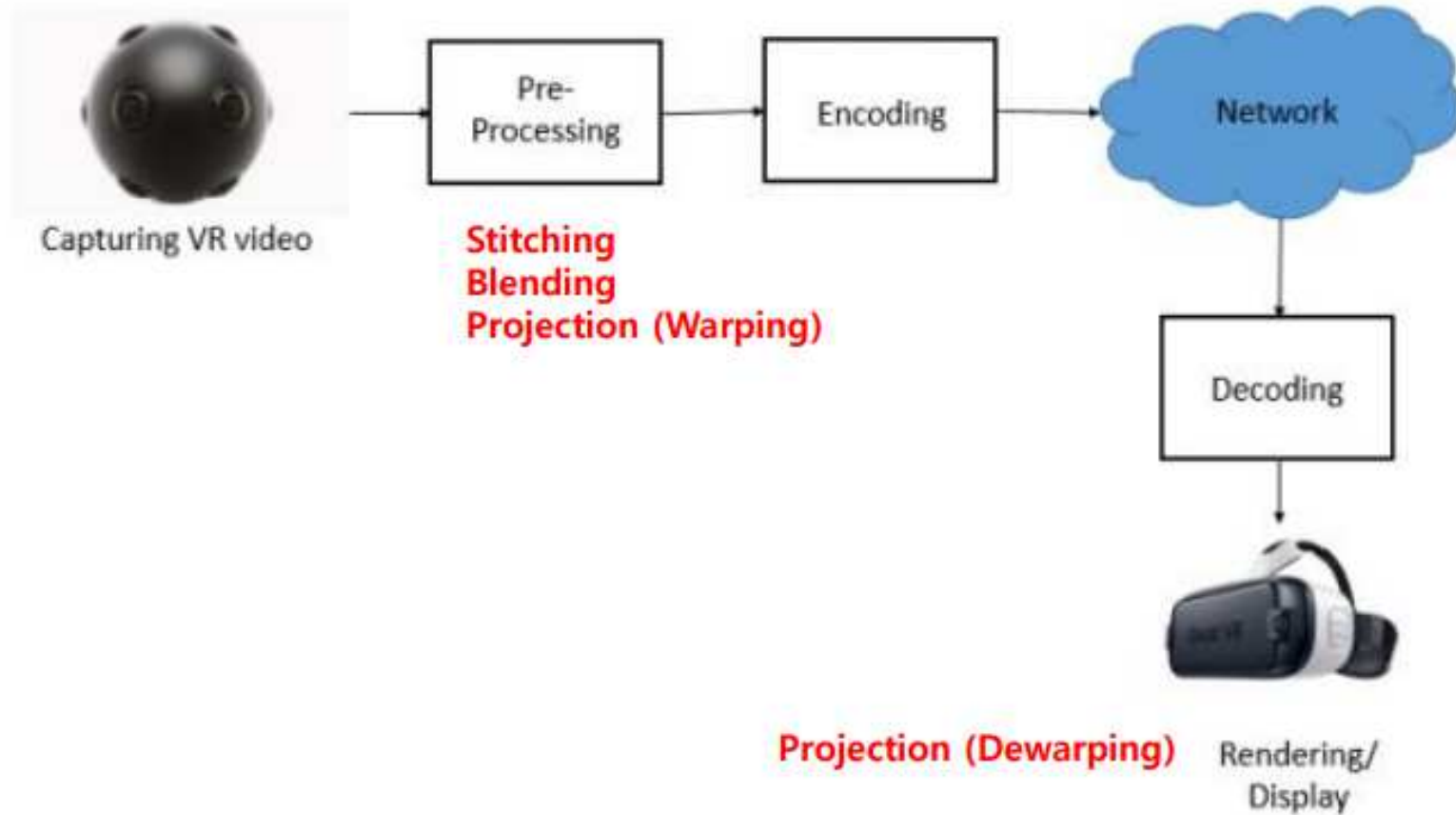
TITAN



Pro

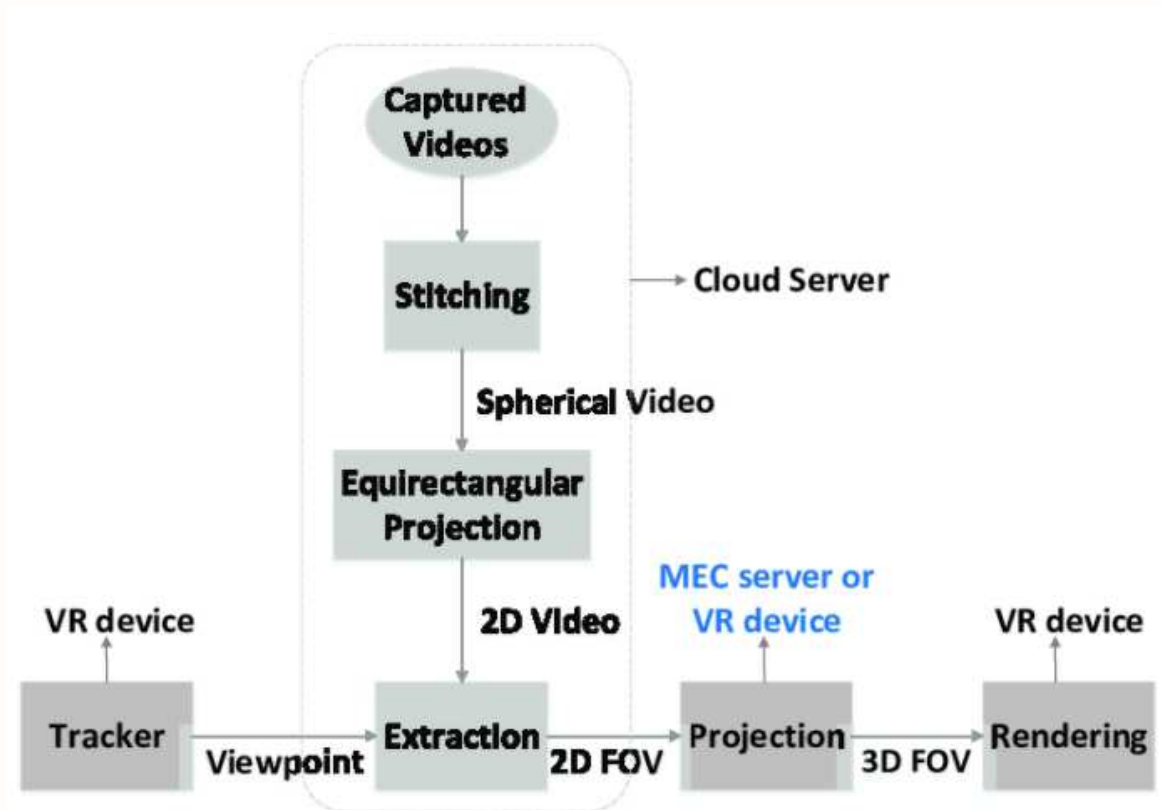
03 360Video

360 Video Processing



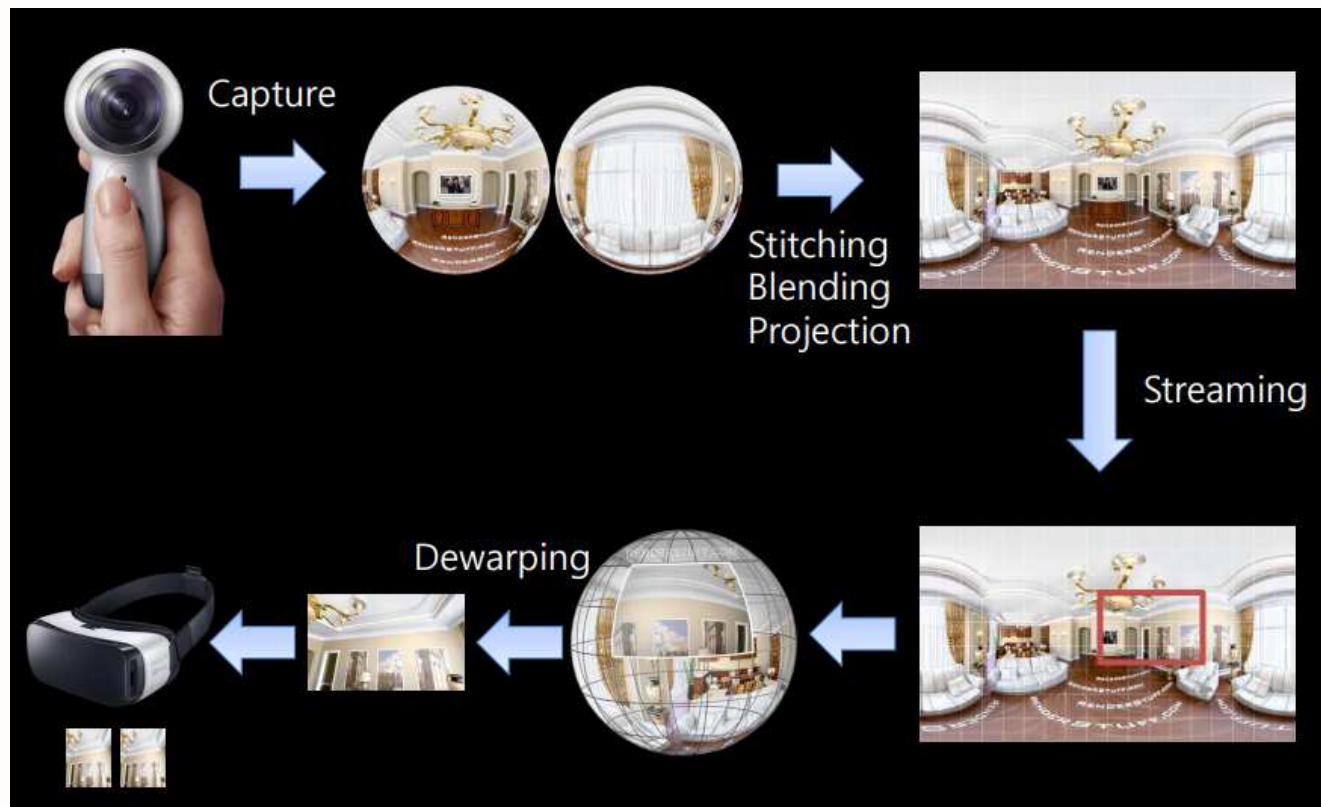


360VR 비디오 제작의 전형적 프레임 워크





360 Video Streaming System (Live 360)

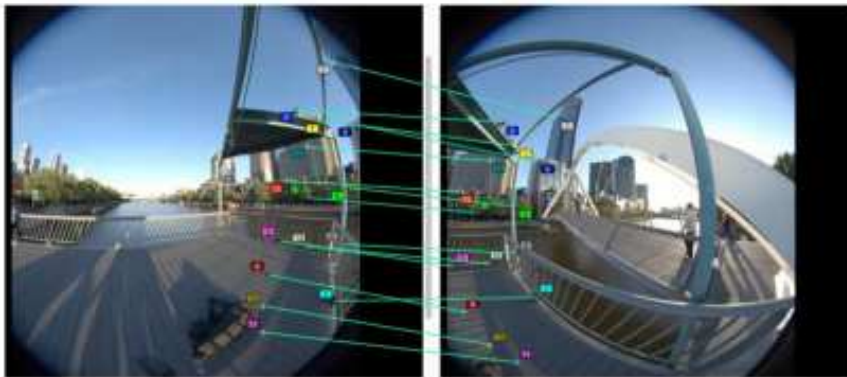




Stitching

Shared features

Unique features appearing in more than one image



Stitched

Images are combined (Seam indicated).





Stitching Error



https://commons.wikimedia.org/wiki/File:Stitching_error.jpg



Blending



(d)



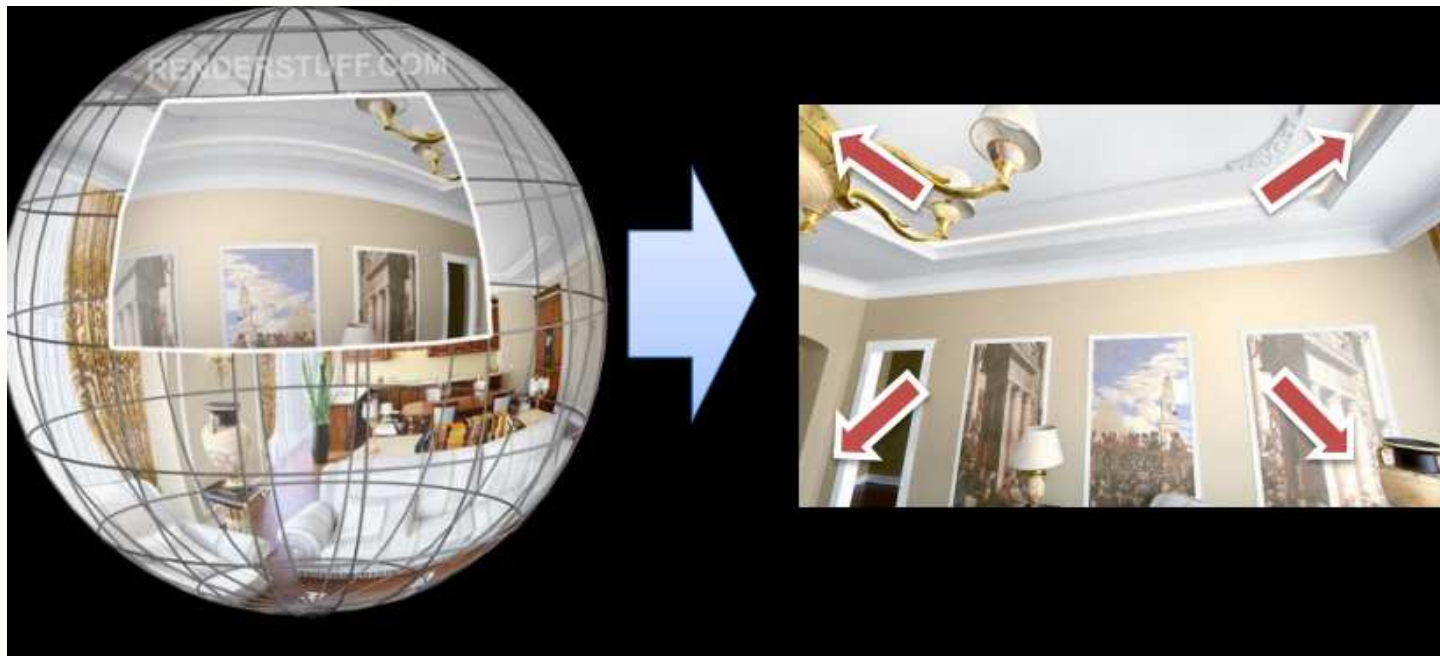
(h)



(i)



Dewarping





Dewarping Error





Timing Mismatch

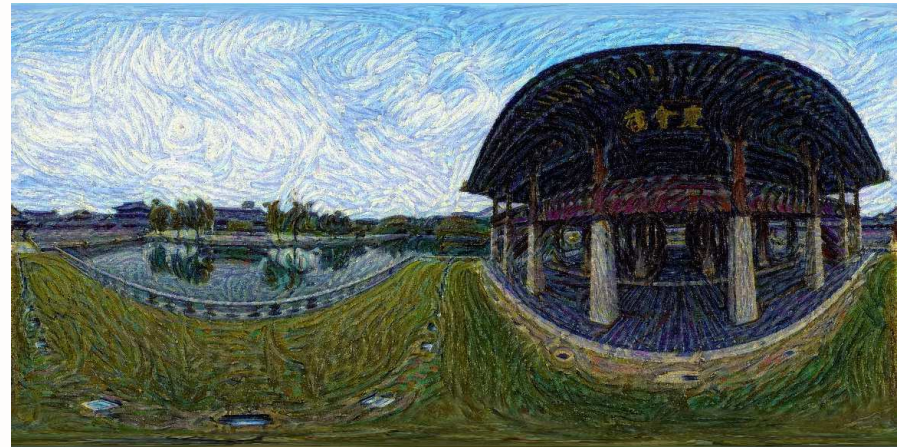
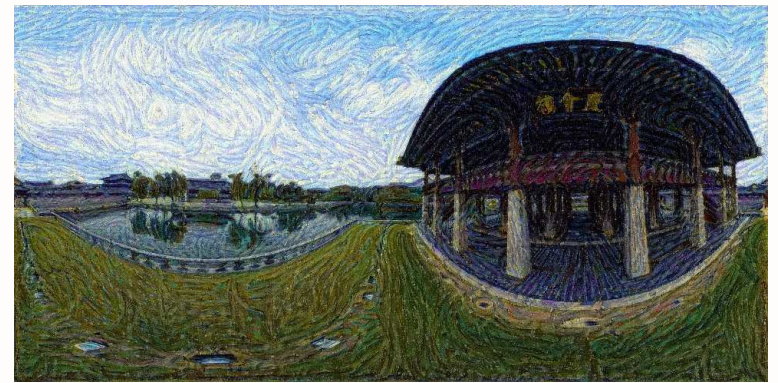


03 360Video

360 Video Processing



Real Project Process



03 360Video

360 Video Processing



Real Project Process



【 2Day 실습 360촬영 및 편집 】

인스타360 ONE X 주요 기능

- FlowState 유동안정화 / 짐벌 없이도 흔들림 없는 영상
- 360도 캡처 / 한번에 찍고 나중에 원하는 부분을 편집하는 신박한 기능
- TimeShift 시네마틱 슬로우모션 + 하이퍼랩스
- Invisible Selfi Stick 셀카봉 지우기
- HDR촬영 / 고급파노라마 스티칭 기술 없어도 360사진 제작
- 초간편 편집 / 모바일 앱으로 모든 편집 가능



04 2일차 실습내용

360 Camera



Insta One X

인스타360 ONE X 제어 앱



Insta360 ONE X - Simple,
snappy 360 photos&video

Insta360

제거

열기

새로운 기능 •

최종 업데이트: 2020. 8. 19.

- 1, You can now migrate files from the ONE X App to the new Insta360 App.
- 2, Routine bug fixes and user experience improvements.



Insta360

Insta360

제거

열기

새로운 기능 •

최종 업데이트: 2020. 9. 15.

- 1, You can now use PureShot for taking photos.
- 2, You can now live stream with ONE R.
- 3, You can now use AquaVision in the app. ...





Insta One X 촬영본 확인

- 1) .insv 파일을 mp4로 변환
 - www.insta360.com 의 상단 메뉴에서 다운로드-Insta360 OneX-STUDIO2020 다운로드
 - 영어버전, next, Install, finish 클릭
- 2) STUDIO2020 열고 Shift로 촬영본 파일들 잡아서 우측 상단의 Export 클릭



Insta360 STUDIO 2020

360 편집 소프트웨어

Insta360 Studio 2020는 사용자가 ONE R, ONE X, EVO, GO, ONE, NanoS, Nano와Air에서 촬영한 동영상과 사진을 편집할 수 있게 해준다. Adobe Premiere Pro용 Insta360 플러그인이 포함되어 있다.

macOS

Windows

LUT

macOS v3.5.1Beta

2020-09-14

다운로드

Updates

1. Added PureShot effect. PureShot's AI algorithm enhances your photo's dynamic range and reduces noise automatically, resulting in a clearer, brighter shot. (ONE R wide angle DNG)

[Read More](#)

v3.5.0

1. You can now use Deep Track. To track, click and drag around a subject in your 360 footage
2. You can now TimeShift by adjusting the playback speed for different sections of the

다운로드

[Read More](#)



트라이포드, 사람 지우기

[에펙]

- 1) 컴포지션에 영상 넣기, 소스선택하여 우클릭 후 Effect-Immersive Video-VR Rotate Sphere 선택
- 2) VR Rotate Sphere 지우기
- 3) 스틸이미지 생성 - ctrl+Alt+s 누르고 output Module을 "JPEG"로 변경하여 Render

[포토샵]

- 1) 스틸이미지 가져와서 레이어 복사
- 2) 3D-Spherical Panorama-New Panorama Layer ~
- 3) 도장 툴 사용해서 트라이포드 및 사람 지우기(Alt클릭 후 문지르기)
- 4) 3D-Spherical Panorama-Export Panorama



트라이포드, 사람 지우기

[에펙]

- 1) 사람제거영상을 소스 위에 얹기, 펜툴 선택해서 따고 소스의 opacity를 40%로 낮춤
- 2) 마스크 Feather값을 50정도로 변경
- 3) 영상과 스틸이미지 선택 후 ctrl+shift+c를 눌러 pre-comp1에 스티커 넣기

감사합니다