특집논문 (Special Paper)
방송공학회논문지 제24권 제2호, 2019년 3월 (JBE Vol. 24, No. 2, March 2019)
https://doi.org/10.5909/JBE.2019.24.2.273
ISSN 2287-9137 (Online) ISSN 1226-7953 (Print)

MPEG-I의 6DoF를 위한 360 비디오 가상시점 합성 성능 분석

김 현 호[a)], 김 재 곤[a)‡]

Performance Analysis on View Synthesis of 360 Videos for Omnidirectional 6DoF in MPEG-I

Hyun-Ho Kim[a)] and Jae-Gon Kim[a)‡]

요 약

360 비디오는 VR 응용의 확산과 함께 몰입형 미디어로 주목받고 있으며, MPEG-I Visual 그룹은 6 자유도(6DoF)까지의 몰입형 미디어를 제공하기 위한 표준화를 진행하고 있다. 제한된 공간내에서 전방위 6DoF를 제공하는 Omnidirectional 6DoF는 제공되는 제한된 수의 360 비디오로부터 임의의 위치에서의 뷰(view)를 제공하기 위한 가상시점 합성이 필요하다. 본 논문에서는 MPEG-I Visual 그룹에서 진행된 전방위 6DoF를 위한 합성에 대한 탐색실험의 성능 및 분석 결과를 기술한다. 즉, 합성하려는 가상시점과 합성을 위한 360 비디오의 입력시점 사이의 거리 및 입력시점의 개수 등의 다양한 실험조건에 따른 합성 성능 결과 및 분석을 제시한다.

Abstract

360 video is attracting attention as immersive media with the spread of VR applications, and MPEG-I (Immersive) Visual group is actively working on standardization to support immersive media experiences with up to six degree of freedom (6DoF). In virtual space of omnidirectional 6DoF, which is defined as a case of degree of freedom providing 6DoF in a restricted area, looking at the scene at any viewpoint of any position in the space requires rendering the view by synthesizing additional viewpoints called virtual omnidirectional viewpoints. This paper presents the performance results on view synthesis and their analysis, which have been done as exploration experiments (EEs) of omnidirectional 6DoF in MPEG-I. In other words, experiment results on view synthesis in various aspects of synthesis conditions such as the distances between input views and virtual view to be synthesized and the number of input views to be selected from the given set of 360 videos providing omnidirectional 6DoF are presented.

Keywords: MPEG-I, Immersive media, 360 video, VR, 6DoF, View synthesis

a) 한국항공대학교 항공전자정보공학부(Korea Aerospace University, School of Electronics and Information Engineering)
‡ Corresponding Author : 김재곤(Jae-Gon Kim)
E-mail: jgkim@kau.ac.kr
Tel: +82-2-300-0414
ORCID: https://orcid.org/0000-0003-3686-4786

※ 이 논문은 2017년도 정부(과학기술정보통신부)의 재원으로 정보통신기술진흥센터의 지원을 받아 수행된 연구임(No. 2017-0-00486).
※ 이 논문의 연구 결과 중 일부는 “2018년 한국방송·미디어공학회 추계학술대회” 및 제 13차 JVET 회의에서 발표한 바 있음.
※ This work was supported by IITP grant funded by the Korea government (MSIT) (No. 2017-0-00486).
※ Parts of this work have been published in the 2018 Fall Conf. of the Korean Institute of Broadcasting and Media Engineers and the 13th JVET Meeting.
· Manuscript received February 1, 2019; Revised March 20, 2019; Accepted March 20, 2019.

I. Introduction

Recently, with the increased commercial interests in deploying Virtual Reality (VR) applications, 360 video has become popular as a new media type giving immersive experiences. In order to enhance immersive experiences with up to six degrees of freedom (6DoF), MPEG-I Visual Group is actively working on standardization supporting 6DoF [1], [2]. In VR space of omnidirectional, which is defined as a case of degree of freedom providing 6DoF in a restricted area, looking at the scene at any viewpoint of any position in space requires rendering by synthesizing additional omnidirectional viewpoint. Such additional rendered viewpoints are called virtual omnidirectional viewpoints. These virtual viewpoints can be synthesized by using texture and depth information from other neighboring viewpoints.

In this paper, we analyze the performance on view synthesis from a set of 360 videos in omnidirectional 6DoF in various ways with different distances and number of input views by using Reference View Synthesizer (RVS) [3], [4]. In the experiments, the dataset of ClassroomImage [5] was used, and the objective quality was evaluated by Equi-Rectangular Projection (ERP) Weighted Spherical PSNR (WS-PSNR) software [6].

II. Test Sequence

The details on the test sequence of ClassroomImage are described in [5]. Figure 1 shows the texture image and

그림 1. ClassroomImage의 첫 번째 프레임의 중앙 뷰(좌: 텍스처 영상, 우: 깊이 영상)
Fig. 1. The first frame of the center view of ClassroomImage (texture image (left) and depth map (right)) [5]

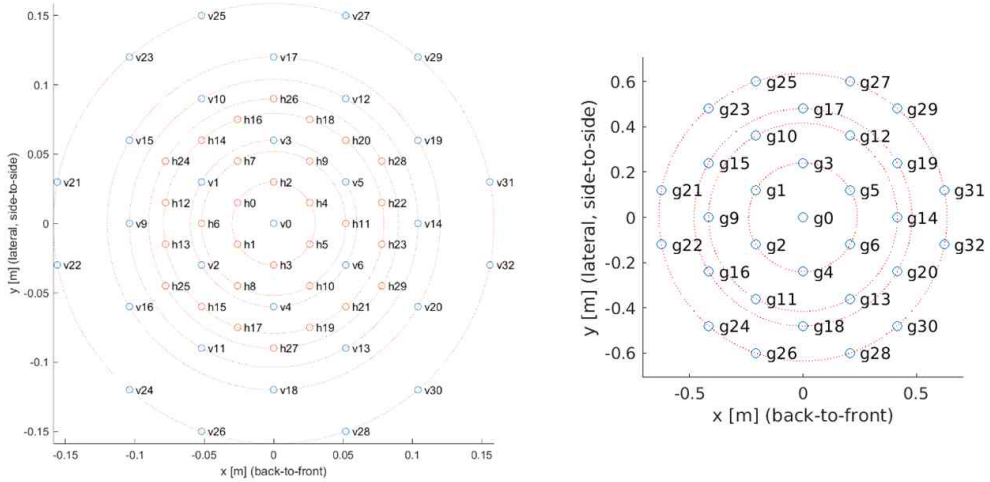

그림 2. ClassroomImage의 뷰 포인트 배치 [5]
Fig. 2. Viewpoint arrangement in ClassroomImage sequence [5]

depth map of the center view (v0), and Figure 2 shows the arrangement of the viewpoints of ClassroomImage, in which 96 viewpoints are distributed within space to provide omnidirectional 6DoF.

III. Experiments on synthesis

1. Methodology

The goal of the experiments in this paper is to check the performance of view synthesis using RVS3.1, according to various synthesis conditions such the distance between the synthesized view and the input view and the number of input views. For this, we performed two experiments as follows:

Measure the performance of synthesized views using two input views with the same distance (60, 120, 240, 480mm) from the input view in terms of the average WS-PSNRs

Measure the performance of synthesized views using different number of input views in terms of the average WS-PSNRs and processing time for the synthesis

The details on each experiment and its performance measured by the ERP WS-PSNR software [6] is given in following sections.

2. View synthesis depending on distance

In this experiment, we synthesize the virtual views by using two input views at the same distances of different cases (60, 120, 240, 480mm) from the input view. Figure 3 and Table 1 show the input viewpoints at the various

표 1. 각 합성 뷰에 대한 60, 120, 240, 480mm 거리의 입력 뷰
Table 1. Input views for view synthesis at the distances of 60, 120, 240, 480mm from each synthesized view [4]

Synthesized view	Input views used for synthesis (used set)			
	60mm (v)	120mm (v)	240mm (g)	480mm (g)
v0 (g0)	v3, v4	v17, v18	g3, g4	g17, g18
v1 (g1)	v2, v10	v11, v25	g2, g10	g11, g25
v2 (g2)	v1, v11	v10, v26	g1, g11	g10, g26
v5 (g5)	v6, v12	v13, v27	g6, g12	g13, g27
v6 (g6)	v5, v13	v12, v28	g5, g13	g12, g28
v9 (g9)	v15, v16	v23, v24	g15, g16	g23, g24
v14 (g14)	v19, v20	v29, v30	g19, g20	g29, g30

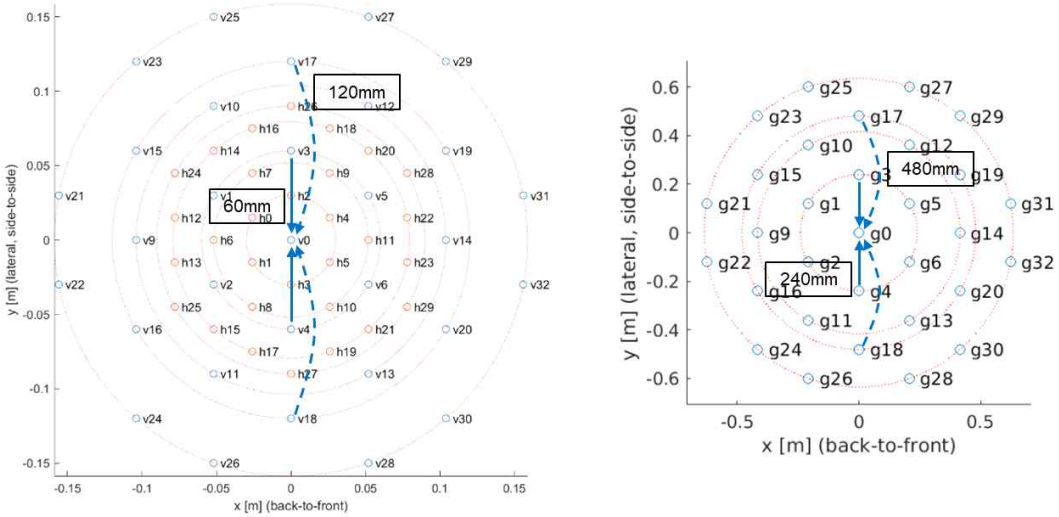

그림 3. v0 또는 g0를 합성하기 위한 60, 120, 240, 480mm 거리의 입력 뷰 포인트 예
Fig. 3. An example of input viewpoints to synthesize v0 or g0 at the distance of 60, 120, 240, 480mm [4]

distances from each synthesized view with the ClassroomImage sequence. We measured the performance of synthesized views with the average WS-PSNR at several positions, each of which is synthesized by corresponding input views with the same distance.

Table 2 shows the performance results of this experiment. There is a significant decrease in the average WS-PSNR when the distance between the input view and the synthesized view increases.

표 2. 다양한 거리의 입력 뷰로부터 합성한 합성 뷰의 WS-PSNR [4]
Table 2. WS-PSNR of the synthesized views at various distances from input views [4]

Synthesized view	Distances from input views (used set)			
	60mm (v)	120mm (v)	240mm (g)	480mm (g)
v0 (g0)	37.92	36.08	32.72	31.34
v1 (g1)	37.90	36.18	32.58	30.83
v2 (g2)	37.98	36.14	32.47	30.71
v5 (g5)	37.90	36.17	32.92	31.64
v6 (g6)	37.95	36.19	32.89	31.69
v9 (g9)	37.61	35.99	32.70	30.36
v14 (g14)	37.63	36.07	32.74	31.51
Avg.	37.93	36.15	32.72	31.15

Figure 4 shows the original g0 image and Figure 5 shows the synthesized g0 position image by using g3, g4. Although it has significant noises, the quality of the images which are synthesized from input views with the distance of 240mm is not bad.

그림 4. g0 원영상
Fig. 4. Original g0 image

그림 5. g3와 g4 입력 뷰를 이용한 g0 위치에서의 합성 영상
Fig. 5. Synthesized image of g0 position by using g3 and g4 input views

3. View synthesis with multiple input views

In this experiment, we synthesize the view with different number of input views to investigate the relationship between synthesis performance and processing time depending on the number of input views. Figure 6 shows the various way to synthesize the virtual view by using different multiple input views with ClassroomImage sequence.

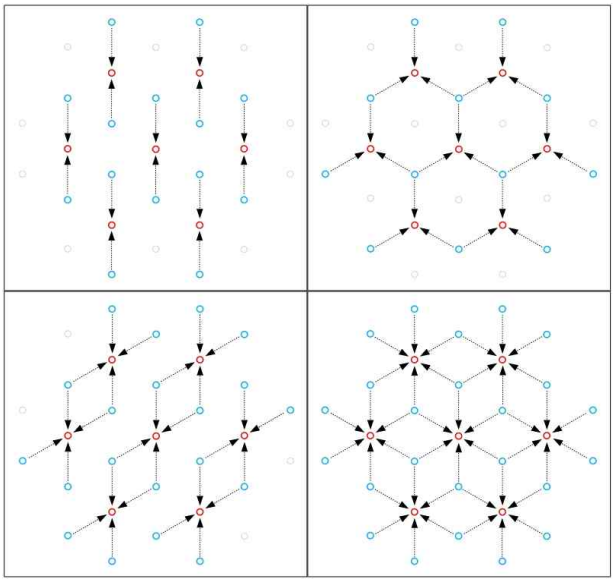

그림 6. 다수의 입력 뷰를 이용한 뷰 합성 예
Fig. 6. Example of view synthesis using multiple input views [4]

We synthesize the view at several positions with the

nearest 2, 3, 4 and 6 input views with the same distance, and calculate WS-PSNR and processing time for synthesis. To eliminate performance differences due to the direction of synthesis, we conducted the experiments only in the same synthesis direction.

In addition, we added g7, g8 each of which is located above and below g0, respectively, to the previous experiment to see how much they have contributed on synthesis performance.

Table 3 shows the WS-PSNR of the synthesized view in each case of using specific number of input view and the processing time required for the synthesis. These results show that as the number of input views to synthesize the virtual view increases, the WS-PSNR and processing time increase. The degree of increase in WS-PSNR is not proportional to the number of input views. However, the processing time increases in proportion to the number of input views.

Table 4 shows the result of adding g7, g8 as input views in the previous experiment. As in the previous experiment, we can see that the processing time increases in proportion to the number of input images. However, it is noted that the performance in terms of WS-PSNR increase significantly when g7, g8 is added to two input views, but other cases is not.

Figure 7 shows the experimental results in Table 3 and 4 in terms of the relation between processing time and WS-PNSR at four different numbers of input views. As mentioned before, both WS-PSNR and processing time increase as the number of input view increase. In addition, when additional views of g7 and g8 are used, both WS-PSNR and processing time are increased which is

표 3. 합성 뷰의 WS-PSNR 및 처리 시간 [4]
Table 3. WS-PSNR and processing time of the synthesized view [4]

	2 inputs		3 inputs		4 inputs		6 inputs	
	WS-PSNR	Time (s)	WS-PSNR	Time (s)	WS-PSNR	Time (s)	WS-PSNR	Time (s)
g0	32.72	23.20	33.28	35.65	33.52	46.42	33.80	74.75
g9	32.30	24.49	32.86	37.35	33.17	48.76	33.45	73.86
g10	32.54	25.71	33.14	42.41	33.51	53.32	33.85	75.38
g11	32.30	24.15	32.89	39.07	33.12	51.09	33.43	78.59
g12	33.14	25.65	33.75	41.33	33.98	56.07	34.22	79.48
g13	32.82	26.24	33.30	39.55	33.62	50.69	33.88	74.83
g14	32.80	25.29	33.30	38.86	33.59	50.69	33.86	70.55
Avg.	32.66	24.96	33.22	39.17	33.50	51.01	33.78	75.35

표 4. 합성 뷰의 WS-PSNR 및 처리 시간 [4]
Table 4. WS-PSNR and processing time of the synthesized view [4]

	2 inputs + (g7, g8)		3 inputs + (g7, g8)		4 inputs + (g7, g8)		6 inputs + (g7, g8)	
	WS-PSNR	Time (s)	WS-PSNR	Time (s)	WS-PSNR	Time (s)	WS-PSNR	Time (s)
g0	33.45	45.32	33.72	57.69	33.81	70.34	33.97	95.60
g9	32.23	49.64	32.68	67.29	32.95	77.75	33.21	98.85
g10	32.73	55.87	33.21	71.60	33.48	98.56	33.77	110.42
g11	32.41	56.05	32.85	66.35	33.02	85.05	33.34	106.78
g12	33.34	54.96	33.77	68.55	33.94	82.52	34.18	108.73
g13	32.98	50.79	33.34	69.90	33.60	73.61	33.80	103.64
g14	33.89	49.08	33.25	62.94	33.46	75.90	33.71	101.41
Avg.	33.00	51.67	33.26	66.33	33.47	80.53	33.71	103.63

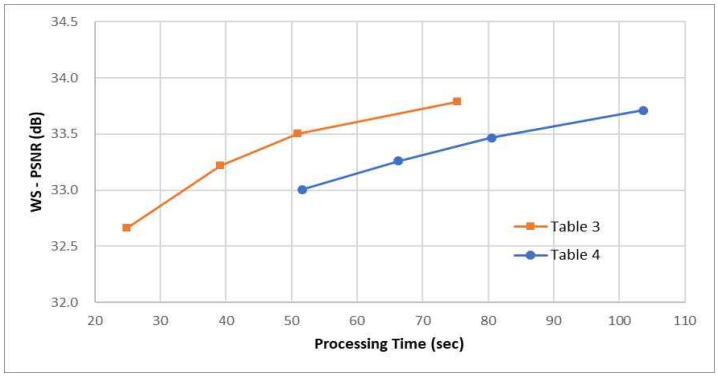

그림 7. 처리 시간과 WS-PSNR 관계 실험 결과(X-축: 처리 시간(sec), Y-축: WS-PSNR (dB)) [4]
Fig. 7. Experiment results on the relation of processing time and WS-PSNR (X-axis: processing time (sec), Y-axis: WS-PSNR (dB)) [4]

shown in the orange-colored line in the Figure 7.

IV. Transmission data amount for synthesis

In this section, we discuss the appropriate number of input images for image synthesis in terms of data amount to be delivered. As the number of input view required for synthesizing a view increases, the data should be delivered

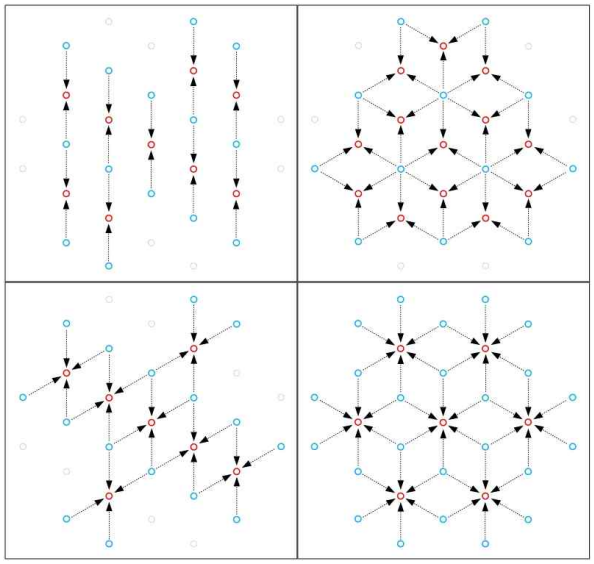

그림 8. 다수의 입력 뷰를 이용한 다양한 합성 방법 [4]
Fig. 8. Various synthesis method using multiple input views [4]

may be increased. Therefore, it may be necessary to make a stable synthesis environment with minimum additional input views for synthesis to be transmitted. Figure 8 shows the view configuration for synthesis using various number of input views. As in the previous experiment, to eliminate the effect of synthesis direction on the performance, we considered the same synthesis direction only.

When synthesizing a virtual view located between two views, it is assumed that the two views are transmitted first and then the virtual view between them is synthesized using the two input views. After that, in order to synthesize the view at a close position, another two input views or one subsequent view need to be transmitted. This means that an average of 1.5 new input views need to be transmitted.

When synthesizing a virtual view with three input views, the associated three views are transmitted first. The three input views can cover the synthesis of virtual view positioned inside of the triangle area enclosed by them, and if the required location of synthesis is out of the triangle area, only one new input view is required to cover another triangle area. It takes more time to synthesize than the case of synthesis using two input images, but it has an advantage in terms of quality of synthesized view and data amount need to be transmitted.

When synthesizing a virtual view with four input views, the four views are transmitted first. The four input views can cover the inside of the rectangular area enclosed by them, and if the required location of synthesis is out of this area, two new input views are required to cover another rectangular area.

When synthesizing a virtual view with six input views, the six views are transmitted first. The six input views can cover the inside of the hexagonal area enclosed by them, and if the required location of synthesis is out of this area, four new input views are required to cover another hexagonal area. This requires a large amount of data to be transmitted, and as we have seen in the previous experiment, it may not be efficient because the quality gain of WS-PNSR is not significant between the cases using four input views and six input views.

V. Conclusions

This paper presents experimental results on virtual view synthesis of omnidirectional 6DoF environment using RVS 3.1 in various way with the distance between synthesized view and input view, and the number of input views.

Based on the experimental results, it is noted that the RVS 3.1 gives performances in the view synthesis below.

Using a closer input view gives better performance than using a farther input view.

Although it has significant noises, the quality of the synthesized images with 240mm away from input views is not bad.

As the number of input view increases, required processing time for the synthesis and objective quality are increase. However, the objective quality is not linearly proportional to the number of input view.

In terms of data amount to be transmitted, using three input views for synthesis would be quite reasonable in case of ClassroomImage sequence.

참 고 문 헌 (References)

[1] “MPEG-I Use Cases for omnidirectional 6DoF, windowed 6DoF, and 6DoF,” ISO/IEC JTC1/SC29/WG11, w16768, Apr. 2017.

[2] J. Jeong, K. Yun, Y. Park, W. Chong, J. Seo, “Proposed architectures for supporting Windowed 6DoF, Omnidirectional 6DoF and 6DoF media,” ISO/IEC JTC1/SC29/WG11, m41555, Oct. 2017.

[3] B. Kroon, G. Lafruit, “Proposed update of the RVS manual,” ISO/IEC JTC1/SC29/WG11, w18068, Oct. 2018.

[4] H.-H. Kim, J.-G. Kim, “Results of the Exploration Experiments for MPEG-I: Omnidirectional 6DoF,” ISO/IEC JTC1/SC29/WG11, m45995, Jan. 2019.

[5] B. Kroon, “ClassroomImage: A frame of ClassroomVideo with less noise and more views,” ISO/IEC JTC1/SC29/WG11, m44762, Oct. 2018.

[6] Y. sun, B. Wang, L. Yu, “WS-PSNR Software Manual,” ISO/IEC JTC1/SC29/WG11, N18069, Oct. 2018.

저 자 소 개

김 현 호

- 2018년 2월 : 한국항공대학교 학사 졸업
- 2018년 3월 ~ 현재 : 한국항공대학교 항공전자정보공학과 석사과정
- ORCID : https://orcid.org/0000-0002-8645-6142
- 주관심분야 : 영상처리, 비디오 코덱, 360 비디오/VR

김 재 곤

- 1990년 2월 : 경북대학교 전자공학과 학사
- 1992년 2월 : KAIST 전기 및 전자공학과 석사
- 2005년 2월 : KAIST 전기 및 전자공학과 박사
- 1992년 3월 ~ 2007년 2월 : 한국전자통신연구원(ETRI) 선임연구원/팀장
- 2001년 9월 ~ 2002년 11월 : Columbia University 연구원
- 2007년 9월 ~ 현재 : 한국항공대학교 항공전자정보공학부 교수
- 2015년 12월 ~ 2016년 1월 : UC San Diego 방문교수
- ORCID : https://orcid.org/0000-0003-3686-4786
- 주관심분야 : 비디오 신호처리, 비디오 부호화 표준, UHD/Immersive Media, 영상통신