강태선
MBC 제작기술국

EXPERT
KNOW-HOW

디포커스 트랙킹을 이용한 가상 스튜디오 실재감 증대 연구 ❷

+ 강태선 MBC 제작기술국

지난 호에서는 가상스튜디오에서의 실재감 향상을 위해 카메라의 디포커스 특성 파라미터 의 정의, 측정 및 분석, 포커스에서 거리와 심도 구하기에 대해 알아보았다. 이번 호에서는 그래픽의 블러링을 이용한 디포커스 효과 구현 및 디포커스 컨트롤러 개발과 가상세트, 실제 세트와의 실재감 주관 평가 결과의 분석을 다루고자 한다.

그래픽의 블러링을 이용한 디포커스 효과 구현

실시간 그래픽에서 구현한 디포커스 기능은 세 가지 문제점이 발견되었다. 첫째, 블러링의 정도가 제한되어 있어 이 양을 초과하여 효과를 줄 수 없으며 둘째, 거리에 따라 블러링의 정도가 변하지 않으며 셋째, 블러링에 사용하는 알고리즘은 통상적인 가우시안 블러링이 아닌 선형 블러링이라는 점이다. 이를 개선하고자 openGL의 Shader 기능을 이용하여 가우시안 블러링을 구현하였다.
가우시안 함수는 자연현상을 가장 잘 표현하는 함수식 중의 하나이다. 다음은 2차원 가우시안 함수이다. 여기서 σ 는 표준 편차이다.

$$G(x,y)=\frac{1}{2\pi\sigma^2}e^{-\frac{x^2+y^2}{2\sigma^2}} \quad (1)$$

2차원 가우시안 함수는 1차원 가우시안 함수로 분리하여 표현하면 아래와 같다.

$$G(x,y)=\frac{1}{2\pi\sigma^2}e^{-\frac{x^2+y^2}{2\sigma^2}}=\frac{1}{\sigma\sqrt{2\pi}}e^{-\frac{x^2}{2\sigma^2}}\frac{1}{\sigma\sqrt{2\pi}}e^{-\frac{y^2}{2\sigma^2}}=G(x)G(y) \quad (2)$$

이 특성을 사용하여 x축과 y축 방향으로 나누어 계산하여 많은 계산량을 줄일 수 있다. 즉 모든 픽셀에 대해 x축 기준으로 가우시안 함수를 적용한 후 그 결과를 가지고 y축 가우시안 함수를 적용하여 계산하면 된다.

openGL Shading language(약어로 GLSL 또는 GLslang)를 이용하여 사용자가 특별한 렌더링 방법을 Shader로 구현할 수 있다. 가상스튜디오용 실시간 그래픽에서는 쉐이더를 구현하기 위한 인터페이스를 제공한다. [그림 1]은 쉐이더 코딩을 위해 제공되는 유저 인터페이스를 보여준다. 여러 번 렌더링이 가능하며 각 렌더링 패스에 대해 코딩이 가능하고 렌더링을 어디에 할 것인지 그것이 입력으로 다시 활용될 것인지 몇 번 반복할 것인지 등을 설정하고 컴파일 버튼을 누르면 가상스튜디오 소프트웨어의 렌더링 엔진과 연결되는 구조이다. Shader는 Vertex shader와 Fragment shader를 코딩한다. Vertex code는 모델뷰나 프로젝션 매트릭스를 이용한 vertex 위치 변환을 하며 vertex 컬러나 라이팅 계산을 위해서 텍스처 좌표 생성과 변환을 한다. Fragment code는 픽셀을 처리하는 특별한 기능으로 컬러

계산, 픽셀당 텍스처 좌표 계산, 포그 계산, vertex의 라이팅이 필요할 경우의 노말 계산 등을 한다. 이 인터페이스를 사용하여 쉐이더를 만들고 각 세팅 값을 설정하고 Vertex code와 Fragment code를 코딩해준다[1].

쉐이더에는 uniform과 varying 변수가 있는데 유니폼 변수는 렌더링 패스에서 일정한 값을 가지며 여러 쉐이더나 openGL 간의 연결(linkage)을 형성한다. varing 변수는 계산된 값을 다른 쉐이더에서 사용할 때 필요하다. 실시간 그래픽의 autofilled 유니폼 변수는 렌더링 엔진에서 사용하는 값을 쉐이더에서 사용할 수 있도록 자동으로 채워지는 변수이다. 대표적인 것으로는 다음과 같다[1].

그림 1. 쉐이더 인터페이스 구조

m3_PassIteration	현재의 iteration(렌더링 패스 수) 넘버이다.
m3_Field	인터레이스 렌더링일 경우 필드를 나타낸다. (0:upper, 1: lower)
m3_ViewMatrix	View matrix이다.
m3_ViewMatrixInverse	Inverse of view matrix 이다.
m3_NearFar	카메라에서 렌더링에 사용하는 Near/Far camera values로 이 범위만 렌더링 한다.
m3_FrameBufferSize	렌더링하는 컬러의 값을 가지는 Frame buffer의 size이다.
m3_FrameBufferColor	렌더링한 결과 픽셀의 컬러값을 가지는 Color frame buffer이다.
m3_FrameBufferDepth	은면 제거에 사용되는 Depth frame buffer이다.

가우시안 블러링을 위해 커널 매트릭스를 만들었다. 이 매트릭스의 값에 따라 첫 번째 렌더링에 수평으로 인접 픽셀과 컨볼루션 하였고 두 번째 렌더링에는 수직으로 컨볼루션을 실행했다. 블러링에 사용되는 매트릭스는 파스칼 트라이앵글을 사용하여 웨이트를 구하였다. [그림 2]는 파스칼 트라이앵글을 이용하여 웨이트를 구하는 도표이다[2].

Index N	Coefficients	Sum = 2^N
0	1	1
1	1 1	2
2	1 2 1	4
3	1 3 3 1	8
4	1 4 6 4 1	16
5	1 5 10 10 5 1	32
6	1 6 15 20 15 6 1	64
7	1 7 21 35 35 21 7 1	128
8	1 8 28 56 70 56 28 8 1	256
9	1 9 36 84 126 126 84 36 9 1	512
10	1 10 45 120 210 252 210 120 45 10 1	1024
11	1 11 55 165 330 462 462 330 165 55 11 1	2048
12	1 12 66 220 495 792 924 792 495 220 66 12 1	4096

그림 2. 파스칼 트라이앵글

그림 3. 실시간 그래픽의 쉐이더를 사용한 가우시안 블러링 적용 영상

강태선
MBC 제작기술국

Vertex code에서는 모델뷰와 프로젝션 매트릭스를 vertex에 적용시킨다. 그리고 텍스처 좌표를 생성한다. Fragment code에서는 렌더링 엔진에서 제공하는 오토필드 유니폼 변수를 사용하고 vertex code에서 가져온 텍스처 좌표와 설정한 블러링 매트릭스 커널의 웨이트를 사용하여 인접 픽셀의 컬러값과 컨볼루션을 실행한다. 커널의 크기는 유니폼 변수로 외부에서 조정이 가능하도록 하였다. [그림 3]은 구현한 쉐이더를 사용하여 실시간 그래픽에 블러링을 적용한 영상이다.

렌즈의 디포커스 파라미터와 그래픽 블러링과의 매칭

렌즈의 특성인 Bdf와 실시간 그래픽 쉐이더의 가우시안 블러링 커널의 배수를 나타내는 스케일 팩터와의 관계를 측정하기 위해 가우시안 커널의 사이즈를 0에서 5배까지 증가시키며 블러링을 적용한 그래픽영상을 분석하여 해당되는 Bdf를 측정하였다. 34.8cm 모니터를 사용했으며 모니터 상수 K=3.16으로 계산되며 모니터의 확대 지수를 MZ라하고 MZ=0.6으로 측정하면

$$Bdf = \frac{loc \times K}{MZ} = \frac{loc \times 3.16}{0.6} \quad (3)$$

로 계산된다. [그림 4]는 차트로 작성한 관계도이다.

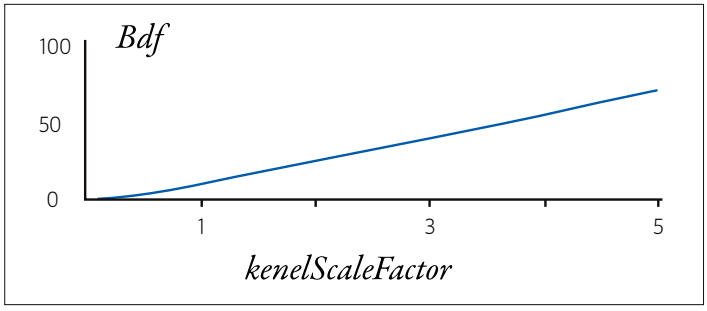

표 4. 커널 스케일 팩터와 Bdf 관계도

[그림 4]을 이용하여 관계식을 도출하면 kernelScaleFactor는

$$kenelScaleFactor \approx \frac{1}{13.5} \times Bdf \quad (4)$$

라고 근접하게 정의된다.

디포커스 컨트롤러(Defcon)의 구현

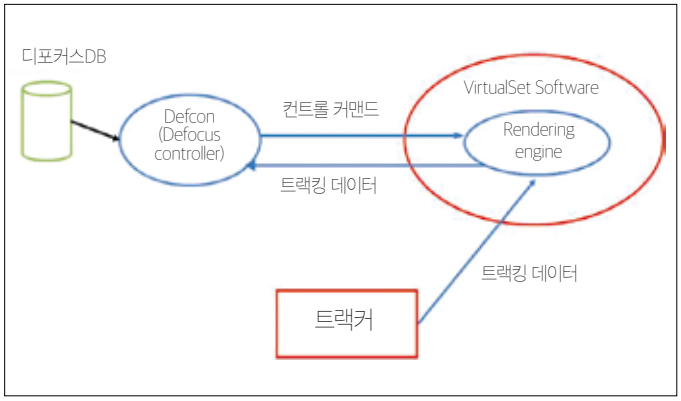

그림 5. Defcon 개념

[그림 5]는 데프콘의 개념이다. 트랙커에서 가상 세트 소프트웨어의 렌더링 엔진으로 매 필드마다 트랙킹 데이터를 전송하며 렌더링 엔진에서는 데이터를 받기 위한 플러그인과 이 데이터를 외부에서 취득할 수 있는 API가 동작된다. Defcon은 렌더링 엔진으로부터 트랙커의 Fen(포커스 인코더 값)과 Fovv(수직 화각)를 읽어온다. 그리고 거리와 Fen의 관계를 정의한 도표에서 거리를 보간하여 구한다. 이 거리는 렌더링 엔진의 디포커스 기능 중 초점이 맞는 거리를 나타내는 DISTANCE를 컨트롤하는 커맨드에 활용된다.

Fovv에서 focal length를 구하고 전후방 심도를 구한다.

렌더링 엔진의 초점 맞는 영역인 RANGE는 전방심도를 사용하기로 하면 $RANGE = \text{전방심도} \times 2$ 로 정의된다. DefocusDB를 활용하여 디포커스 정도인 APPLY_FACTOR 값을 계산한다. 3가지 값 즉 DISTANCE, RANGE, APPLY_FACTOR를 컨트롤하는 커맨드를 렌더링 엔진으로 보낸다. 블러링 연동을 위해 Defcon 기능을 업데이트하였다. [그림 6]과 같이 카메라로부터 Set Distance 거리에 가상 세트가 위치해 있고 초점이 맞는 지점의 끝부터 Virtual Set까지의 거리를 Max Blur Distance라 하고 이때의 Blur의 양을 Max Blur라 정의한다. 그러면 거리에 따른 디포커스는 측정한 결과로 리니어하다고 보면 다음과 같은 선형식으로 가상 세트의 blurring 정도를 구할 수 있다.

$$Blur = \frac{setDistance - (DISTANCE + RearDOF)}{MaxBlurDistance} \times MaxBlur \quad (5)$$

만약 $SetDistance - (DISTANCE + RearDOF) > MaxBlurDistance$ 이면

$$Blur = MaxBlur,\ Blur \geq 0$$

Defcon에 SetDistance, MaxBlur, MaxBlurDistance 인터페이스를 추가하고 입력된 값과 계산된 DISTANCE, RANGE 값을 사용하여 실시간으로 Blur의 양을 계산하여 실시간 그래픽 블러링 Shader의 Kernel Size를 컨트롤하는 명령을 보내주면 된다. 블러링 인터페이스를 추가한 Defcon GUI는 [그림 7]과 같다.

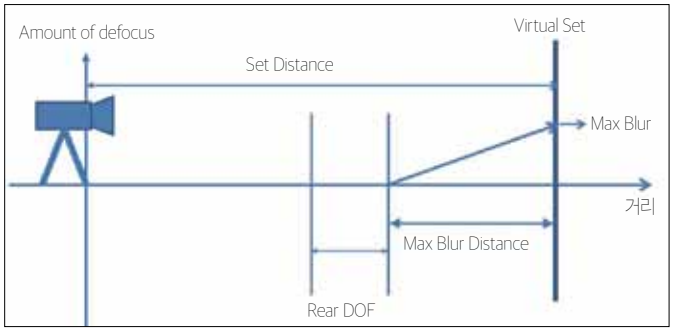

그림 6. 가상 세트 블러링을 위한 개념

그림 7. 블러링 인터페이스 추가한 Defcon GUI

가상 세트와 실제 세트와의 실재감 주관 평가

드라마 세트와 근접하게 디자인한 가상 세트와의 특수 연출 샷의 비교분석 실험을 위해 MBC 아침드라마 "내손을 잡아"(연출 : 최은경 카메라: 김태형, 조명: 오승만) 중에 사용된 일식집 세트를 사용하였다. 3DMax에서 실제 도면을 참조하여 모델링 및 텍스처 작업을 하였

강태선
MBC 제작기술국

고, scene은 실시간 그래픽 제조사의 3DMax 플러그인으로 실시간 그래픽 파일로 변환하였다. 가상 세트를 실시간 렌더링하여 [그림 8]과 같이 실제 세트와 아주 유사함을 확인하였다.

주관 평가를 위하여 제작한 카메라 샷은 아래 세 가지이다.

① 풀 피규어 샷에서 타이트 바스트 샷까지 줌인하기.

② 세트 배경에 포커스인하다가 전방의 인물로 포커스 이동하기.

③ 거리 차이가 있는 2명의 인물 오버 샷에서 먼 곳의 세트에 포커스 인하고 뒤쪽 인물 그리고 앞쪽 인물로 포커스 이동하기.

주관 평가 과정은 두 개의 트랙으로 이루어지며 실제 세트에서 가상 세트를 제작, 촬영하는 과정과 실제 세트에서 평가용 샷을 촬영하는 과정으로 이 두 개의 촬영 및 편집이 끝나면 이 영상들을 모아 하나의 평가용 영상으로 편집한다. [그림 9]는 평가용 실제 세트 선정부터 주관 평가 분석까지의 과정이다.

(a) 실제 세트 (b) 가상 세트

그림 8. 주관 평가용 실제 세트와 가상 세트

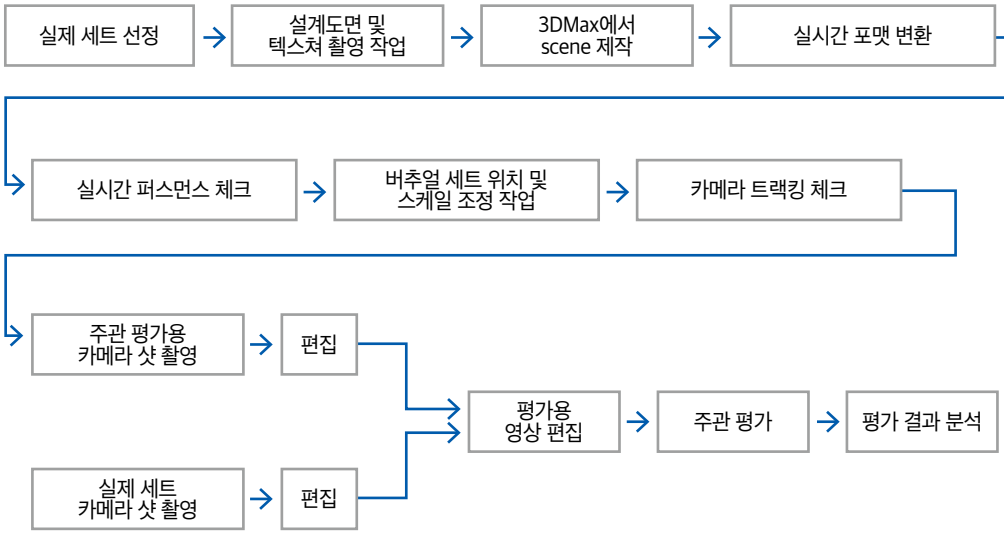

그림 9. 주관 평가 과정

[그림 10과 11]은 블러링 있는 가상 세트 인물 두 명 포커스 이동 샷과 실제 세트 인물 두 명 포커스 이동 샷의 샘플 영상이다.

(a) 포커스 파

(b) 뒤쪽 인물 포커스

(c) 앞쪽 인물 포커스

그림 10. 블러링 있는 가상 세트 인물 두 명 포커스 이동 샷

(a) 포커스 파

(b) 뒤쪽 인물 포커스

(c) 앞쪽 인물 포커스

그림 11. 실제 세트 인물 두 명 포커스 이동 샷

강태선
MBC 제작기술국

실재감 주관 평가 결과 분석

영상의 품질을 측정하는 방법에는 객관적, 주관적 평가 방법이 있다. 주관적 평가 방법은 인간의 인지적 특성을 반영하고 있어 객관적 평가보다 좀 더 신뢰할 수 있는 방법이다[3]. 관찰자 그룹은 방송 전문가 그룹과 일반인 그룹으로 분류하여 각각 10명과 5명에 대해 평가를 실시했다. 평가할 항목은 첫째는 가상 세트에서 동일한 카메라 샷에 대해 디포커스 효과를 적용한 것과 적용하지 않은 것과의 실재감 차이에 대한 평가이고 둘째는 실제 세트와 가상 세트 간의 실재감에 대한 평가이다.

모니터 A와 B에 나타난 영상을 보고 A에 5점을 주고 B에 3점을 주었다면, A의 실재감이 B보다 2점이 더 선호된다고 분석한다. 반대로 A에 3점을 주고 B에 5점을 주었다면 A의 선호도는 0이 된다. 다음 모든 관찰자의 선호도를 합계하여 평균을 구한다. [식 6]은 선호도 평균을 구하는 식이다.

$$K=\frac{1}{15}\sum_{i=1}^{15}|A_i-B_i| \quad (6)$$

[그림 12]는 여러 가지 비교 영상별 선호도 평균치를 보여 준다. 평가 결과를 분석하면 가상 세트에 블러링을 준 것은 안 준 것에 비해 매우 현격한 실재감의 증대 효과를 보여준다. 줌인이 들어가는 V1의 경우에는 상대적으로 적은 실재감 증대를 보여주는데 이것은 카메라 샷의 특성상 심도를 깊은 렌즈를 사용했을 경우에 포커스 아웃이 되지 않은 경우를 상정한 결과인 것으로 보인다. 실제 세트와 가상 세트에 블러링을 준 것과의 선호도는 별반 차이를 보이지 않는다. 즉 실제 세트 영상과 가상 세트에 블러링을 준 것이 별로 차이가 없는 실재감을 나타내며 즉 실제 세트와 가상 세트의 구별이 어렵다는 것으로 분석된다.

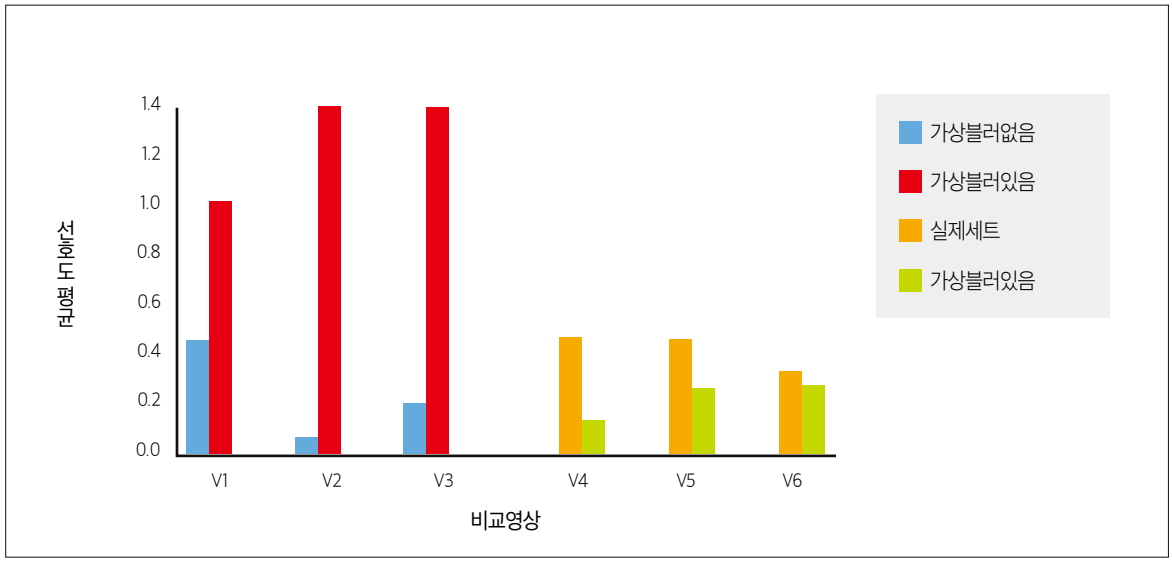

그림 12. 비교영상별 선호도 평균

[그림 13]은 방식별 선호도 평균으로 가상 세트에 블러링을 준 것이 안준 것에 비해 약 4.7배의 실재감 향상이 되었음을 보여준다. 실제 세트와 비교하면 가상 세트가 약 52%의 실재감이 만족됨을 보여준다. 방송관계인의 경우만 보면 약 15배의 실재감 향상을 표시했고 가상 세트가 87%의 실재감 만족도를 보여 실제 세트와 가상 세트가 거의 구별을 할 수 없을 정도임을 보여준다.

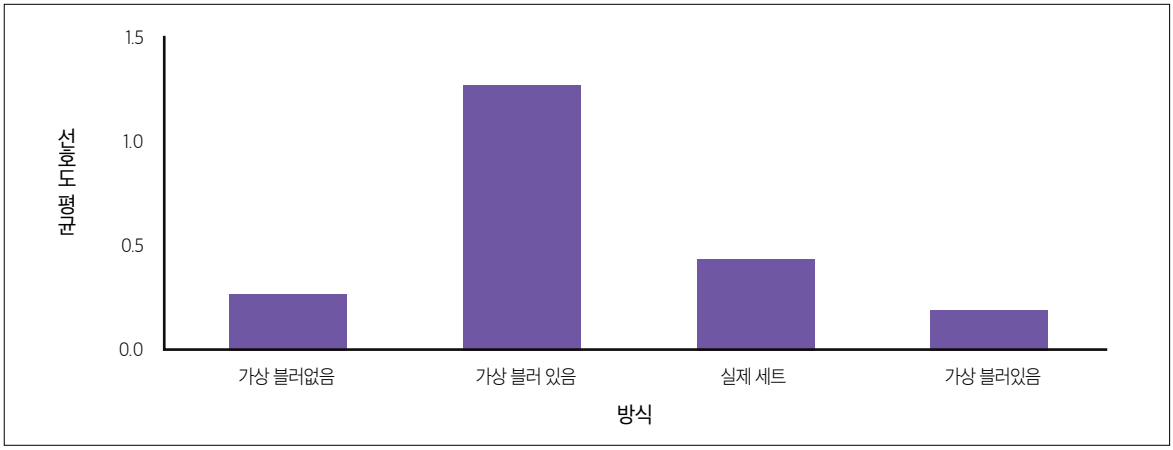

그림 13. 방식별 선호도 평균

결론

카메라 렌즈의 심도 특성은 방송 드라마에서 인물의 감정이나 스토리텔링의 미묘한 부분을 연출하기 위해 사용되지만 가상스튜디오 제작 분야에서는 크게 중요시되는 부분은 아니었다. 인물과 세트간의 실재감을 증대시키기 위해서는 그래픽의 버추얼 카메라 심도를 실제 카메라 렌즈의 특성과 유사하게 구현하여 포커스인 되는 영역과 아웃되는 영역을 구분하여 렌더링해주는 기술이 필요하다. 이렇게 함으로써 가상스튜디오가 드라마와 같은 창의적인 카메라 샷이 요구되는 분야에도 적용될 것으로 본다.

본 연구에서는 가상스튜디오 실재감의 이슈들 중에 카메라 심도의 구현을 위하여 기존의 디포커스 기술을 조사하였고, 그 제한성을 파악하고 이를 개선하고자 제안한 카메라 렌즈의 디포커스 파라미터 특성에 적응적으로 연동되는 그래픽 효과를 생성하고 합성하여 마치 블루룸에 가상 세트가 실재처럼 존재하는 듯 한 시각 효과를 창출하고자 하였다. 이를 위해 포커스 값에 따라 초점을 맞춘 피사체와의 물리적 거리를 측정하는 실험을 하였다. 측정 데이터를 이용하여 실시간으로 카메라 트랙킹 센서에서 구한 포커스 데이터로 피사체간의 물리적 거리를 계산하였고 심도를 구하는 수식을 활용하여 전후방 심도를 구하였다. 카메라 렌즈의 디포커스 정도를 분석하기 위하여 디포커스 파라미터들을 제안하고 실험을 통하여 그 값들을 추출하였다. 추출된 값을 분석하여 여러 유의미한 정보를 알아내었고, 이를 적응적으로 실시간 가상 세트 소프트웨어에 적용하고자 그래픽의 블러링 정도와 실제 카메라 영상의 디포커스 파라미터가 일치하도록 매칭시키고 그래픽 디포커스의 기능을 정의하는 변수들을 외부에서 컨트롤 해주는 Defcon(defocus controller)을 개발하여 운용함으로 가상스튜디오의 실재감을 향상시켰다.

또한, 실시간 그래픽의 openGL Shader기능을 이용하여 가우시안 블러링을 구현하는 쉐이더를 제작하였고 Defcon에서 이 쉐이더의 기능을 컨트롤하여 렌즈로부터 초점을 맞춘 거리와 심도, 포커스인 영역을 벗어난 거리에 따라 블러링의 정도가 트랙킹 되도록 하였다. 실제 세트와 가상 세트에 블러링을 적응적으로 적용한 영상과의 주관적 실재감 평가를 실시하여 실재감이 증대된 것을 검증하였다.

* [참고문헌]

[1] Brainstorm, "BrainStorm eStudio reference guide" http://www.brainstorm.es

[2] http://rastergrid.com/blog/2010/09/efficient-gaussian-blur-with-linear-samling

[3] 신종희, "SD-to-HD 방송영상 Upscaling시 Interpolation 방식에 따른 장르별 화질 분석", 연세대학교 공학대학원 통신.방송 공학 전공 석사학위논문, 2013.