반복 가산 기법을 이용한 Fresnel 홀로그램의 고속 계산 알고리듬

정회원 최 현 준*, 종신회원 서 영 호**, 김 동 욱*

Fast Computation Algorithm of Fresnel Holograms Using Recursive Addition Method

Hyun-Jun Choi* ***Regular Member***, Young-Ho Seo**, Dong-Wook Kim* ***Lifelong Members***

요 약

디지털 홀로그래픽 비디오 시스템을 제작하기 위해서는 디지털 홀로그램을 가능한 빠르게 생성하는 것이 중요하다. 본 논문에서는 디지털 홀로그램의 전체 좌표를 대상으로 반복적인 가산 연산을 이용하여 Fresnel 홀로그램의 생성 속도를 높이는 알고리듬을 제안한다. 디지털 홀로그램을 계산하기 위한 3차원 객체는 컴퓨터 그래픽(computer graphic, CG)으로 제작한 깊이영상(depth-map image)을 이용하였다. 본 논문에서 제안하는 알고리듬은 부동소수점 형식의 반복가산기법을 이용하여 디지털 홀로그램의 위상을 고속으로 계산하는 기법이다. 실험결과 제안한 알고리듬은 일반적인 CGH 수식을 이용한 기법의 70%, [3]에서 제안한 기법보다 30%이상 연산속도가 빨라졌다.

Key Words : Digital holography, Computer generated holography, Digital hologram, 3D, High-speed

ABSTRACT

For digital holographic video system, it is important to generate digital hologram as fast as possible. This paper proposed a fixed-point method and fast generation method that can calculate the Fresnel hologram using operation of whole-coordinate recursive addition. To compute the digital hologram, 3D object is assumed to be a collection of depth-map point generated using a PC. Our algorithm can compute a phase on a hologram by recursive addition with fixed-point format at a high speed. When we operated this algorithm on a personal computer, we could maximally compute digital hologram about 70% faster than conventional method and about 30% faster than of [3]'s method.

Ⅰ. 서 론

가장 이상적인 3차원 영상 디스플레이 기법으로 주목 받고 있는 홀로그래피는 주로 미국, 유럽, 일본 등에서 활발한 연구가 진행되고 있다. 특히, 실시간으로 홀로그래피 영상을 재생하는 홀로그래픽 비디오는 차세대 3DTV를 위한 핵심기술로 주목 받고 있다.

CGH는 Brown과 Lohmann에 의해 1966년에 제안되었다[1]. CGH기법은 광학 신호들을 근사화(approximation)한 후 PC상에서 수학적인 연산으로 간섭패턴(interference pattern 혹은 디지털 홀로그램)을 얻을 수 있는 기법이다. CGH기법을 이용할 경우 실제 공간상의 객체 혹은 가상의 객체로부터 손

※ 이 논문은 2008년도 교내 학술연구비 지원에 의해 연구되었음.
* 광운대학교 전자재료공학과 ({chj, dwkim}@kw.ac.kr), **광운대학교 교양학부 (yhseo@kw.ac.kr)
논문번호 : KICS2008-01-038, 접수일자 : 2008년 1월 18일, 최종논문접수일자 : 2008년 4월 7일

쉽게 디지털 홀로그램을 얻을 수 있다. 하지만, CGH기법으로 디지털 홀로그램을 계산하기 위해서는 상당한 연산량(약 1㎝×1㎝×1㎝의 3차원 객체(3D object)를 공간상에서 재생하기 위해 일반적인 PC를 이용할 경우 약 900초가 소요됨.)이 요구된다. 이를 개선하기 위해 Yoshikawa[2]는 3차원 객체와 생성할 디지털 홀로그램의 거리의 차만을 반복적으로 더해서 연산속도를 높이는 기법을 제안하였다. 하지만 [2]의 기법은 에러 누적과 많은 승산연산이 필요하다는 단점이 있다. [3]은 생성될 디지털 홀로그램의 전체 좌표에서 이루어지던 일반적인 CGH연산을 x-축의 첫 번째 좌표((0,0), (1,0), (2,0), (3,0) ..... (N,0))에서만 수행하고 나머지 좌표의 연산은 첫 번째 연산 결과에 미리 계산된 값과 이전 연산 결과를 가산하는 기법을 제안하였다.

MIT[4,5]에서는 일반적인 CGH 연산식의 각 항들을 미리 계산해 놓은 LUT(Look-up Table)을 이용하는 기법을 제안하였다. 하지만 이런 기법들은 많은 하드웨어 자원을 사용한다. 최근에는 DSP, FPGA, 그래픽 카드(GPU)등을 이용하여 연산속도를 높이는 기술들도 소개되고 있다[6-11]. 하지만, 본 논문에서는 LUT, 병렬처리(parallel processing), 파이프라이닝(pipelining) 기법 등을 이용한 하드웨어 기반의 기술들에 대해서는 다루지 않고, CGH기법의 연산횟수를 줄여 고속연산을 가능케 하는 알고리듬을 제안한다.

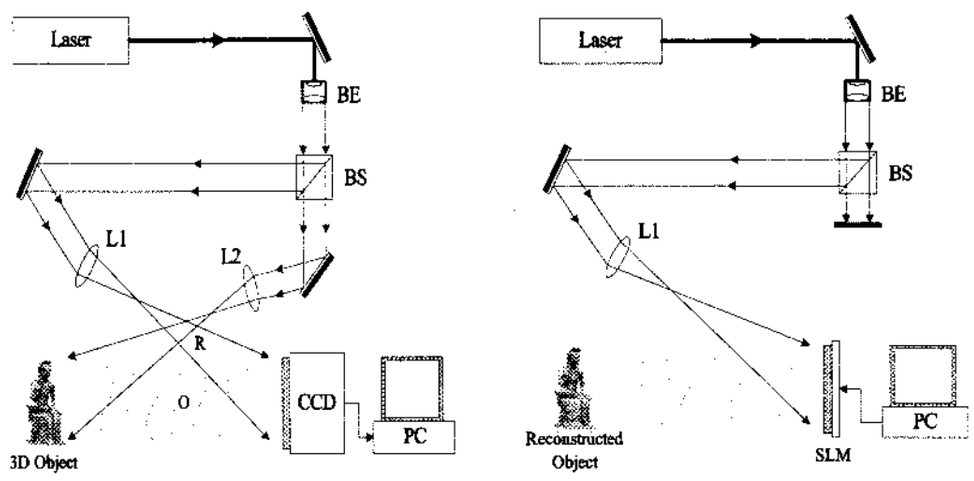
본 논문에서는 [3]에서 제안한 반복가산(recursive addition) 기법을 분석하여 3차원 객체영상과 디지털 홀로그램 좌표들 사이의 규칙성을 찾아낸 후, 디지털 홀로그램의 첫 번째 x-축 좌표들에서 수행되는 전연산(whole operation)과 전체 좌표에서 수행되는 부분 곱셈(partial multiplication) 및 부분 덧셈(partial addition)의 횟수를 줄였다.

본 논문의 Ⅱ장에서는 일반적인 CGH 기법과 [3]에서 제안한 기법에 대해 설명하고, Ⅲ장에서는 이전에 제안된 기법들을 분석한 후 새로운 알고리듬을 제안한다. Ⅳ장에서 실험결과를 보이고, Ⅴ장에서 결론을 맺는다.

Ⅱ. 컴퓨터 생성 홀로그램

2.1 디지털 홀로그램 및 CGH 기본이론

디지털 홀로그램은 광학장비 대신 전자장비들을 이용하는 방식으로, 홀로그래피의 간섭무늬를 CCD 카메라에 기록하고 비디오 신호로 전송하여 수신단



(a) 기록 (b) 복원
(a) recoding (b) reconstruction

그림 1. 디지털 홀로그램
Fig. 1. Digital hologram

에서 SLM(Spatial Light Modulator)에 표시된 간섭무늬에 레이저광을 조사(illumination)함으로써 영상을 재생하는 기법이다. 그림 1에서 디지털 홀로그램의 시스템 구성을 보이고 있다. 디지털 홀로그램은 기존의 광학 홀로그램에 의한 기법과 동일하게 레이저광을 집광 렌즈(condensing lens)로 평행광(collimated wave)을 만들고, 빔 분리기(beam splitter)로 참조파(reference wave)와 객체파(object wave)로 나눈다. 객체파는 객체에 조사된 다음 참조파와 직접 CCD에 조사되어 간섭무늬를 형성한다. 간섭무늬 정보를 SLM에 인가하고 여기에 평행광을 조사하면 1차 회절광(1'st diffraction beam)이 발생하여 실사 영상을(real image) 재생할 수 있다[12].

홀로그램은 그림 1과 같은 광학 시스템을 이용하여도 취득할 수 있지만 이러한 광학 시스템 자체를 수학적으로 모델링한 연산에 의해서 취득할 수도 있다. 이러한 수학적인 연산을 통해 얻어진 홀로그램을 CGH라고 한다. 여러 종류의 CGH가 있지만 본 논문에서는 "Phase" 방식의 계산 방법을 사용한다. 즉, 그림 1(a)에서 객체에서 CCD로 입사되는 파에서 위상 성분만을 이용하여 홀로그램을 생성하는 것이다.

2.2 CGH 계산식

본 절에서는 일반적인 CGH 연산 기법과 [3]에서 제안한 위상을 이용한 반복 가산방식의 연산 기법을 설명한다.

일반적인 CGH생성 식은 (1)과 같이 정의된다.

$$I_\alpha = \sum_j^N A_j \cos\left(k\sqrt{(px_\alpha - px_j)^2 + (py_\alpha - py_j)^2 + z_j^2}\right) \quad (1)$$

여기서 α와 j는 홀로그램과 3차원 객체, k는 참조파

의 파수(wave number)로 $2\pi/\lambda$로 정의되고, p는 홀로그램의 화소 크기(pixel pitch), x_α와 y_α는 홀로그램의 좌표, x_j, y_j, 및 z_j는 3차원 객체의 좌표를 나타낸다.

그림 2에서는 CGH 기법을 적용하기 위한 3차원 객체와 디지털 홀로그램의 좌표 배열을 보이고 있다. 그림 2에서 보이고 있는 좌표 배열은 2×2크기의 3D 객체로 4×4크기의 디지털 홀로그램을 생성할 때의 예이다. 이때, 디지털 홀로그램을 생성하기 위해서는 식 (1)의 연산을 2×2×4×4=64회 수행해야 한다.

식 (1)을 Taylor 전개(Taylor expansion) 후 첫 번째 항으로 근사화를 시키면 다음과 같이 정리할 수 있다.

$$I_\alpha = \sum_j^N A_j \cos\left(\frac{2\pi}{\lambda}\left(Z_j + \frac{p^2}{2Z_j}\left((x_\alpha - x_j)^2 + (y_\alpha - y_j)^2\right)\right)\right) \quad (2)$$

식 (2)의 항들을 정리하면 다음과 같다.

$$I_\alpha = \sum_j^N A_j \cos\left(2\pi(\theta_z + \theta_H)\right) \quad (3)$$

$$\left(\theta_z = \frac{z_j}{\lambda}, \theta_H = \frac{p^2}{2\lambda z_j}\left(\acute{x}_{\alpha j} + \acute{y}_{\alpha j}\right)\right)$$

여기서 $x_{\alpha j}$와 $y_{\alpha j}$는 $(x_\alpha - x_i)$ 와 $(y_\alpha - y_i)$를 의미한다.

디지털 홀로그램에서의 한 점 $(x_\alpha + x_i, y_\alpha)$에서의 위상 $\theta_H(x_{\alpha j}+n, y_{\alpha j}, z_j)$는 다음 식과 같이 표현할 수 있다.

$$\begin{aligned}&\theta_H(x_{\alpha j}+n, y_{\alpha j}, z_j) \quad (4)\\ &= \frac{p^2}{2\lambda z_j}\left((x_{\alpha j}+n)^2 + \acute{y}_{\alpha j}\right)\\ &= \frac{p^2}{2\lambda z_j}\left(\acute{x}_{\alpha j} + \acute{y}_{\alpha j}\right) + \frac{p^2}{2\lambda z_j}\left(2nx_{\alpha j} + n^2\right)\\ &= \theta_H(x_{\alpha j}, y_{\alpha j}, z_j) + \Gamma_{xn}\end{aligned}$$

여기서 Γ_{xn}은 다음과 같이 정의한다.

$$\Gamma_{xn} = \frac{p^2}{2\lambda z_j}\left(2nx_{\alpha j} + n^2\right) \quad (5)$$

식 (5)에 n값들을 대입하면, n=1일 때 Γ_{x1}은,

$$\Gamma_{x1} = \frac{p^2}{2\lambda z_j}\left(2x_{\alpha j} + n^2\right) \quad (6)$$

n=2일 때 Γ_{x2}는,

$$\begin{aligned}\Gamma_{x2} &= \frac{p^2}{2\lambda z_j}(4x_{\alpha j}+4) \quad (7)\\ &= \frac{p^2}{2\lambda z_j}(2x_{\alpha j}+1) + \frac{p^2}{2\lambda z_j}(2x_{\alpha j}+1)\\ &\quad + \frac{p^2}{2\lambda z_j}\times 2\\ &= \Gamma_{x1} + \Gamma_{x1} + \Delta_x\end{aligned}$$

여기서 Δ_x는 다음과 같이 정의한다.

$$\Delta_x = \frac{p^2}{2\lambda z_j}\times 2 \quad (8)$$

다시, n=3일 때 Γ_{x3}을 계산해보면,

$$\begin{aligned}\Gamma_{x3} &= \frac{p^2}{2\lambda z_j}(6x_{\alpha j}+9) \quad (9)\\ &= \frac{p^2}{2\lambda z_j}(4x_{\alpha j}+4) + \frac{p^2}{2\lambda z_j}(2x_{\alpha j}+1)\\ &\quad + \frac{p^2}{2\lambda z_j}\times 4\\ &= \Gamma_{x2} + \Gamma_{x1} + 2\Delta_x\end{aligned}$$

즉, n=N일 때 Γ_{xN}은 다음과 같이 일반화 시킬 수 있다.

$$\Gamma_{xN} = \Gamma_{x(N-1)} + \Gamma_{x1} + (N-1)\Delta_x \quad (10)$$

식 (10)에서 Γ_{x1}과 Δx값을 미리 계산해 놓으면 디지털 홀로그램의 x축 첫 번째 좌표 값들만을 식 (2)를 이용하여 계산하고 $x+1$번째부터는 Γ_{x1}, Δx, 그리고 이전에 계산된 $\Gamma_{x(N-1)}$값을 가산함으로써 CGH 연산을 수행할 수 있다. 이때, 식 (2)와 같은 CGH 연산을 전연산, 동일한 x축 상의 홀로그램 값들을 연산할 때 사용되는 식 (10)의 일부 연산들을 부분승산과 부분가산이라고 정의한다. 위에서 설명한 [3]의 기법을 그림 3에서 보이고 있다. 그림에서 명암으로 처리한 블록은 전연산을, 일반블록은 부분연산된 결과임을 의미한다.

예를 들어, 일반적인 CGH 연산은 200×200[pixel^2]의 3차원 객체를 대상으로 1,024×1,024[pixel^2]의 디지털 홀로그램을 생성할 때 전연산이 1,024×1,024×200×200=41,943,040,000번 반복 수행된다. 하지만, [3]에서 제안한 기법은 전연산이 1,024×200×200=40,960,000번 반복 수행되고, 부분승산과 부분가산은 1,023×1,024×200×200=41,902,080,000

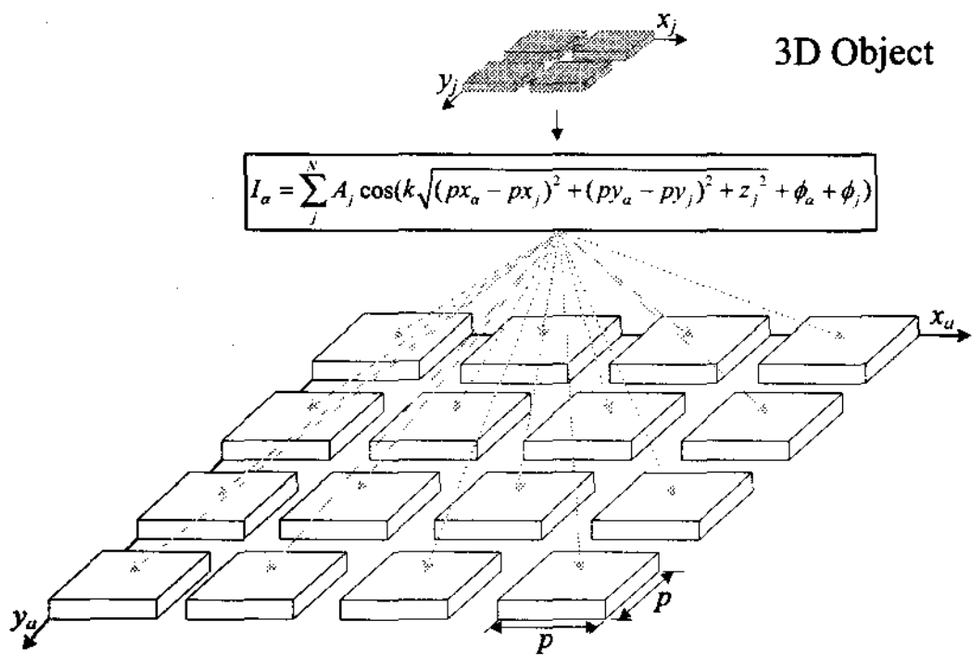


그림 2. 일반적인 CGH 알고리듬
Fig. 2. Conventional CGH algorithm

번 수행된다. 즉, 복잡한 전연산을 반복적인 부분승산과 부분가산 연산으로 대체함으로써 전연산의 횟수를 99.9%이상 감소시키는 기법이다.

Ⅲ. 제안한 고속 CGH 알고리듬

3.1 고정 소수점 연산

본 절에서는 고속의 CGH 연산을 위해 PC의 성능과 비용 등의 제약이 고려된 수 체계 정밀도 분석을 수행하였다. 수 체계 정밀도 분석의 대상은 식 (3)과 (8)의 θ_x, Δ, π, *Cos*항들이다. 이 항들의 비트수 제약이 연산 결과에 미치는 영향을 분석하였고, 이를 바탕으로 고속의 CGH연산 알고리듬을 제안한다. 그림 4에서 고정 소수점(fixed-point) 시뮬레이션을 통한 PSNR변화 결과를 보이고 있다. 시뮬레이션 결과는 C++ 언어를 사용하여 100개의 3차원 CG영상에 대한 평균값을 산출하였다. 고정 소수점 연산을 위한 소수점의 절사(truncation) 기준은 HVS(human visual system)를 고려하여 PSNR값이 30dB이상이 되는 점으로 하였다. 그림 4를 보면

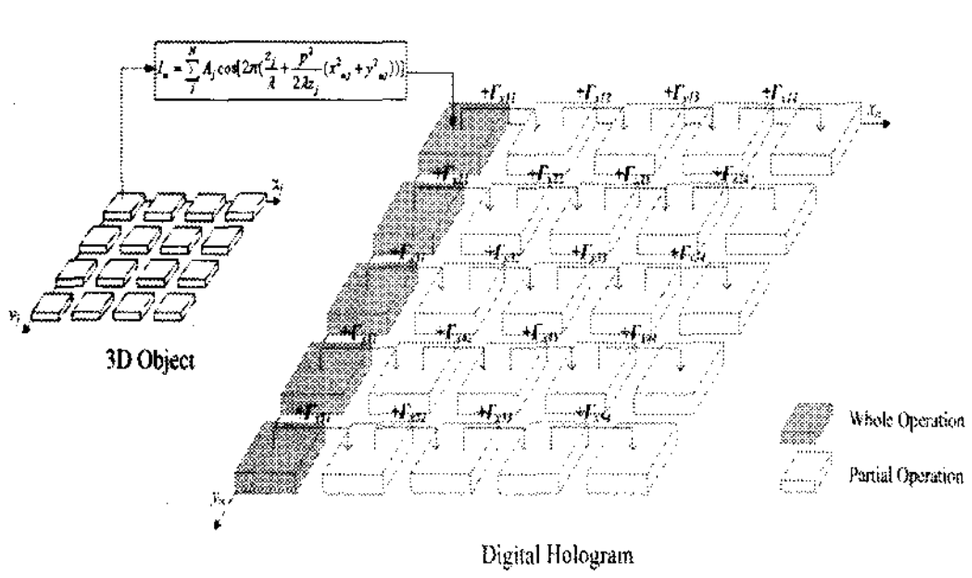


그림 3. 반복 가산을 이용한 CGH 알고리듬
Fig. 3. CGH algorithm using recursive addition

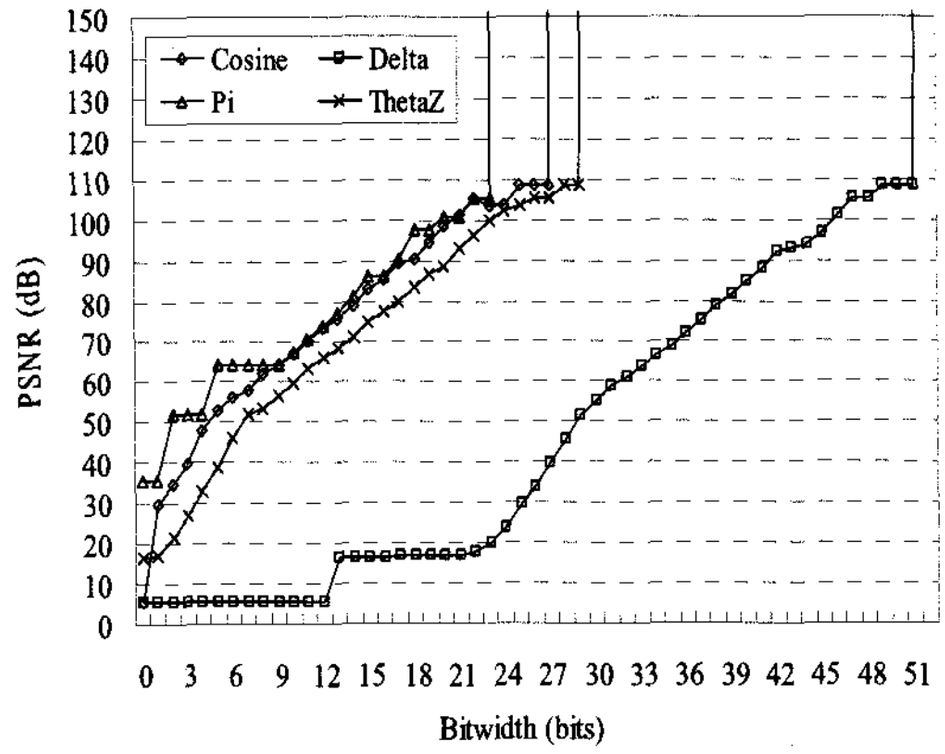


그림 4. 고정소수점 연산을 위한 시뮬레이션
Fig. 4. Simulation result for fixed-point computation

*Cos*함수는 소수점 3번째, Δ는 소수점 26번째, θ_x는 소수점 5번째, 그리고 π는 소수점 이하를 모두 없앤 후 정수 연산만으로도 30dB가 넘는 결과를 보이고 있다. 본 논문에서는 이 점들을 고정소수점 연산을 위한 비트넓이(bit-width)로 정한다.

3.2 제안한 CGH 알고리듬

2.1절에서 설명하였듯이 [3]에서 제안한 CGH 기법은 3차원 객체의 모든 점들과 홀로그램 좌표의 수만큼 전연산을 수행했던 일반적인 기법에 비해 향상된 연산기법이다. 본 절에서는 [3]의 기법을 개선하여 디지털 홀로그램의 x-축과 y-축 방향의 전체 좌표에서 이뤄지는 반복 가산 알고리듬을 제안한다.

3.2.1 디지털 홀로그램의 x축 방향 반복 가산

식 (10)의 Γ_{x1}과 Δx항에 원래의 수식을 대입하여 정리하면 식 (11)과 같다.

$$\Gamma_{xN} = \frac{p^2}{2\lambda z_j}(2nx_{\alpha j} + 1) + \Gamma_{x(N-1)} + \frac{2p^2}{2\lambda z_i}(N-1) \quad (11)$$

식 (11)을 보면 3차원 객체의 한 점을 홀로그램의 첫 번째 x-축((0,0) 좌표로 시작하는 x-축) 상에서 계산할 때 Γ_{x2}이후의 값들을 변하게 하는 변수는 N뿐임을 알 수 있다. 즉, 3차원 객체의 한 점을 연산할 경우 홀로그램의 첫 번째 x-축에서 연산되었던 Γ_{xN}값들은 다른 x-축((0,1)~(0,n))의 Γ_{xN}값들과 동일하다는 것을 알 수 있다. 따라서, 첫 번째 x-축 홀로그램들을 연산할 때 계산하였던 Γ_{xN}값들은 다른 x-축 홀로그램들을 계산할 때 그대로 이용할 수 있다.

앞서 설명한 기법은 그림 5와 같이 3차원 객체의

동일한 열(column)에 존재하는 점들을 대상으로 CGH 연산을 수행할 경우로 확장할 수 있다. 하지만, 동일한 열에 존재하는 점들을 식 (11)의 수식으로 반복 가산할 때 x_j, x_α, N값들은 같지만 z_j는 변하기 때문에 먼저 계산되었던 Γ_{xN}값들을 다시 이용할 수 없다. 이를 해결하기 위해 식 (11)을 다음과 같이 정리할 수 있다.

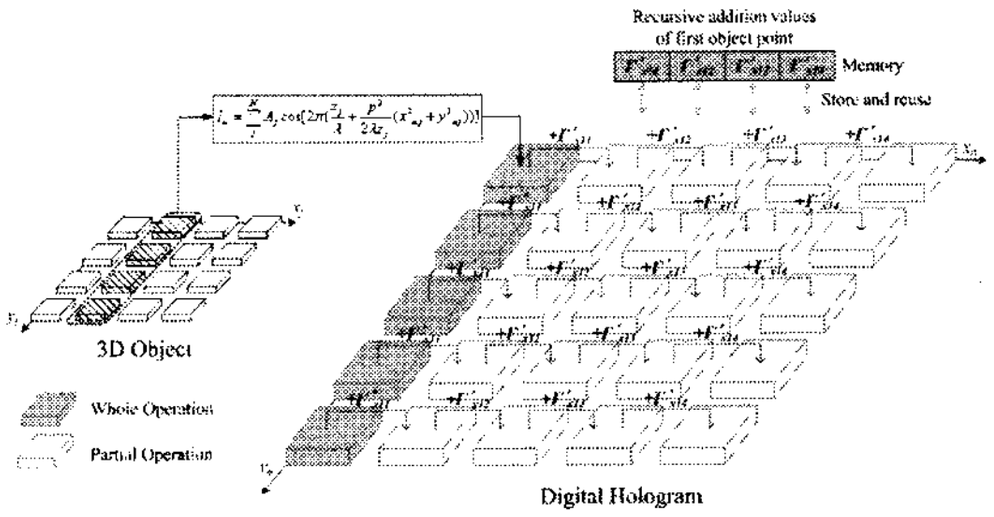


그림 5. 동일한 3차원 객체 열들의 x-축 반복가산 기법을 이용한 CGH 알고리듬
Fig. 5. CGH algorithm using x-axis recursive addition of same 3D object column

N=1일 때, Γ'_{x3} 은,

$$\Gamma'_{x1} = \frac{1}{z_j}\left(\frac{p^2}{2\lambda}(2x_{\alpha j}+1)\right) = \frac{1}{z_j}(\Gamma_1) \tag{12}$$

N=2일 때, Γ'_{x2} 는,

$$\begin{aligned}\Gamma'_{x2} &= \frac{1}{z_j}\left(\frac{p^2}{2\lambda}(2x_{\alpha j}+1)\right) \\ &\quad + \frac{1}{z_j}\left(\frac{p^2}{2\lambda}(2x_{\alpha j}+1)\right) + \frac{1}{z_j}\left(\frac{2p^2}{2\lambda}\right) \\ &= \frac{1}{z_j}(\Gamma_{x1}+\Gamma_{x1}+\Delta_x)\end{aligned} \tag{13}$$

N=3일 때, Γ'_{x3} 은,

$$\begin{aligned}\Gamma'_{x3} &= \frac{1}{z_j}\left(\frac{p^2}{2\lambda}(2x_{\alpha j}+1)\right) + \frac{1}{z_j}\left(\frac{p^2}{2\lambda}(2x_{\alpha j}+1)\right) \\ &\quad + \frac{p^2}{2\lambda}(2x_{\alpha j}+1) + \frac{2p^2}{2\lambda} + \frac{1}{z_j}\left(\frac{2p^2}{2\lambda}\right) \\ &= \frac{1}{z_j}(\Gamma_{x1}+\Gamma_{x1}+2\Delta_x)\end{aligned} \tag{14}$$

n=N일 때 Γ'_{xN}은 다음과 같이 일반화 시킬 수 있다.

$$\Gamma'_{xN} = \frac{1}{z_j}(\Gamma_{x(N-1)}+\Gamma_{x1}+(N-1)\Delta_x) \tag{15}$$

식 (10)과 (15)에서 확인할 수 있듯이 같은 열의 3차원 객체들을 CGH연산할 때 이용되는 Γ'_{xN} 값들은 z_j를 제외하면 동일하다. 즉, 3차원 객체의 동일한 열에 존재하는 점들을 대상으로 x-축 반복 가산 CGH 연산을 할 경우 z_j를 제외한 나머지 항들은 첫 번째 점의 Γ'_{xN}값들을 그대로 이용하여 연산 시간을 줄일 수 있다.

2.2절에서 설명하였던 [3]의 x-축 반복 연산기법을 디지털 홀로그램의 y-축 방향으로 적용해 보면 다음과 같이 정리할 수 있다.

3.2.2 디지털 홀로그램의 y축 방향 반복 가산

디지털 홀로그램에서의 한 점 $(x_\alpha, y_\alpha+n)$에서의 위상 $\theta_H(x_{\alpha j}+n, y_{\alpha j}, z_j)$를 다음과 같이 표현할 수 있다.

$$\begin{aligned}&\theta_H(x_{\alpha j}, y_{\alpha j}+n, z_j) \\ &= \frac{p^2}{2\lambda z_j}(x_{\alpha j}+(y_{\alpha j}+n)^2) \\ &= \frac{p^2}{2\lambda z_j}(x_{\alpha j}^2+y_{\alpha j}^2)+\frac{p^2}{2\lambda z_j}(2nx_{\alpha j}+n^2) \\ &= \theta_H(x_{\alpha j}, y_{\alpha j}, z_j)+\Gamma_{yn}\end{aligned} \tag{16}$$

여기서 Γ_{yn} 은 다음 식과 같이 정의한다.

$$\Gamma_{yN} = \frac{p^2}{2\lambda z_j}(2nx_{\alpha j}+n^2) \tag{17}$$

n=1일 때 Γ_{y1} 은,

$$\Gamma_{y1} = \frac{p^2}{2\lambda z_j}(2y_{\alpha j}+1) \tag{18}$$

n=2일 때 Γ_{y2} 는,

$$\begin{aligned}\Gamma_{y2} &= \left(\frac{p^2}{2\lambda z_j}(4x_{\alpha j}+4)\right) \\ &= \frac{p^2}{2\lambda z_j}(2y_{\alpha j}+1)+\frac{p^2}{2\lambda z_j}(2y_{\alpha j}+1)+\frac{p^2}{2\lambda z_j}\times 2 \\ &= (\Gamma_{y1}+\Gamma_{y1}+\Delta_y)\end{aligned} \tag{19}$$

여기서 Δ_y는 다음과 같이 정의한다.

$$\Delta_y = \frac{p^2}{2\lambda z_j}\times 2 \tag{20}$$

다시, n=3일 때 Γ_{y3}을 계산해보면,

$$\begin{aligned}\Gamma_{y3} &= \left(\frac{p^2}{2\lambda z_j}(6x_{\alpha j}+9)\right) \\ &= \frac{p^2}{2\lambda z_j}(4y_{\alpha j}+4)+\frac{p^2}{2\lambda z_j}(2y_{\alpha j}+1)+\frac{p^2}{2\lambda z_j}\times 4 \\ &= \Gamma_{y2}+\Gamma_{y1}+2\Delta_y\end{aligned} \tag{21}$$

n=N일 때 Γ_{yN}은 다음과 같이 일반화 시킬 수 있다.

$$\Gamma_{yN} = \Gamma_{y(N-1)}+\Gamma_{y1}+(N-1)\Delta_y \tag{22}$$

식 (10)과 (22)를 비교해 보면 두 식이 같다는 것을 확인할 수 있다. 즉, [3]에서 제안한 x-축 방향 반복 가산 CGH 연산 기법은 y-축 방향으로 확장해서 적용할 수 있다.

3차원 객체의 동일한 열에 위치하는 점들 사이의 규칙성을 찾기 위해 식 (22)에 실제 값들을 대입하여 Γ_{yN}을 계산해 보면 표 1과 같다. 표 1에서는 3차원 객체의 동일한 열에 존재하는 점들의 좌표((1,0), (1,1), (1,2), (1,3))를 대상으로 디지털 홀로그램의 (0,0)으로 시작하는 첫 번째 열을 연산할 경우를 예로 들고 있다. 여기서 δ는 $p^2/2\lambda z_j$를 의미한다. 표 1을 보면 계산된 Γ_{yN}값들은 3차원 객체의 동일한 열에 존재하는 점들의 y-좌표에 따라 규칙적으로 변하는 것을 알 수 있다. 이런 규칙성은 식 (23)과 같이 일반화 시킬 수 있다.

$$\Gamma_{yN} = 2ym+(2y_{\alpha j}+3) \tag{23}$$

표 1과 식 (23)이 의미하는 바는, 3차원 객체의 동일한 열에 위치하는 점들을 대상으로 y-축 방향 반복 가산 CGH 연산을 수행할 경우 홀로그램의 첫 번째 좌표((0,0))에서 계산된 값에 식 (23)으로 계산된 값들을 반복 가산하면 된다. 즉, 3차원 객체의 첫 번째 동일한 열에 존재하는 점들을 연산할 때 계산하였던 Γ_{yN}들은 두 번째 이후 점들의 연산에 그대로 사용하여 연산 시간을 줄일 수 있다.

3.2.3 제안한 전좌표 반복 가산방식의 CGH 알고리듬

본 절에서는 앞서 제안하였던 홀로그램의 x, y-축 방향 반복 가산기법들을 동시에 CGH 알고리듬을 제안한다. 제안한 알고리듬은 다음과 같은 순서로 수행된다.

① 3차원 객체의 동일한 열에 위치하는 점들 중에서 첫 번째 점으로 디지털 홀로그램의 첫 번째 횡(row)에 대해 x-축 방향 반복 가산방식의 CGH 연산을 수행한다.
② 1의 연산시 계산된 $\Gamma_{x1}\sim\Gamma_{xN}$값들을 저장한다.
③ 디지털 홀로그램의 두 번째 횡의 첫 번째 좌표는 첫 번째 횡의 첫 번째 좌표를 기준으로 y-축 방향 반복가산방식의 CGH연산을 수행한다.
④ 3의 연산시 계산된 $\Gamma_{y1}\sim\Gamma_{yN}$값들을 저장한다.
⑤ 디지털 홀로그램의 두 번째 횡의 두 번째 좌표연산부터는 2에서 저장했던 값들을 순서대로 가산한다.
⑥ 동일한 열에 위치하는 3차원 객체의 두 번째 점을 2와 4에서 저장했던 값들로 x, y-축 방향 반복 가산을 수행한다.
⑦ 3차원 객체의 다음 열에 위치하는 점들을 대상으로 1~6의 연산을 반복한다.

그림 6에서는 제안한 3차원 객체의 동일한 열에

표 1. 3차원 객체의 동일한 열에 위치하는 점들의 ΓyN값의 예

Table 1. Example of ΓyN values in the point of same column on 3D object

First point of hologram line (α) : (0,0)					
3D object point(j)		(1,0)	(1,1)	(1,2)	(1,3)
Γ_{yN}	Γ_{y1}	δ	-δ	-3δ	-5δ
	Γ_{y2}	δ+δ+2δ=4δ	-δ-δ+2δ=0	-3δ-3δ+2δ=-4δ	-5δ-5δ+2δ=-8δ
	Γ_{y3}	δ+4δ+4δ=9δ	-δ+0+4δ=3δ	-3δ-4δ+4δ=-3δ	-5δ-8δ+4δ=-9δ
	Γ_{y4}	δ+9δ+6δ=16δ	-δ+3δ+6δ=8δ	-3δ-3δ+6δ=0	-5δ-9δ+6δ=-8δ
	Γ_{y5}	δ+16δ+8δ=25δ	-δ+8δ+8δ=15δ	-3δ-0+8δ=5δ	-5δ-8δ+8δ=-5δ
	Γ_{y6}	δ+25δ+10δ=3δ	-δ+15δ+10δ=2	-3δ+5δ+10δ=12δ	-5δ-5δ+10δ=0

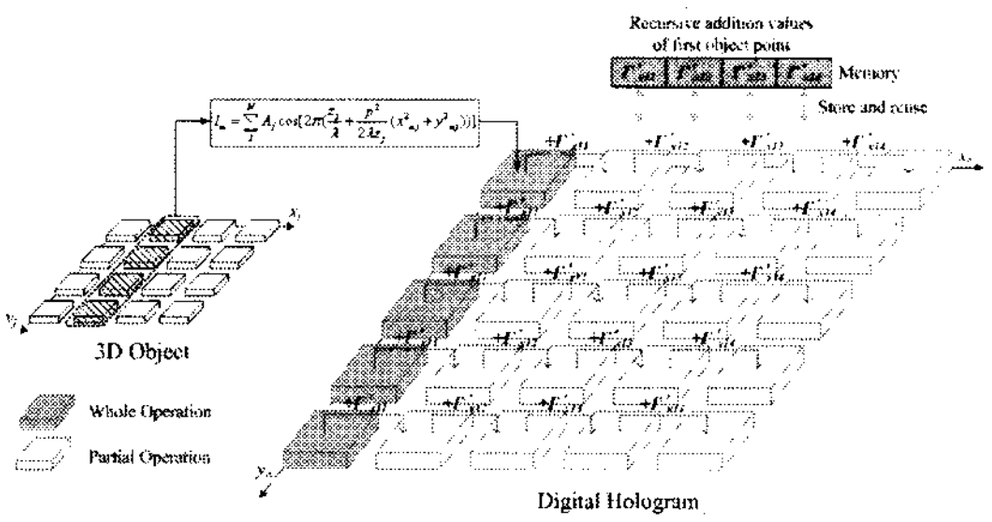


그림 6. 동일한 3차원 객체 열들의 전체 좌표 반복가산 방식을 이용한 CGH 알고리듬
Fig. 6. CGH algorithm using whole-coordinates recursive addition of same 3D object column.

존재하는 점들의 홀로그램 전체 좌표 반복가산 CGH알고리듬을 보이고 있다. 제안한 알고리듬은 전연산을 3차원 객체의 점들의 수만큼만 수행하고 나머지 홀로그램 좌표의 값들은 반복가산 기법을 이용하여 부분연산으로 처리하여 연산 속도를 높인다.

Ⅳ. 실험결과

제안한 알고리듬을 검증하기 위해 표 2와 같은 환경에서 식 (1)과 같은 일반적인 CGH연산과 2.2절에서 설명한 [3]의 기법, 그리고 본 논문에서 제안한 홀로그램 전체 좌표 반복가산 알고리듬을 구현하였다.

3차원 객체는 CG로 생성한 200×200의 깊이정보 영상을 사용하였고, 각 점의 빛의 강도는 깊이정보 영상의 화소 값으로 대체하였다. 생성할 디지털 홀로그램은 복원환경을 고려하여 적색(red) 채널, 1,024×1,024 크기의 회색조(8-bit) 영상으로 설정하였다.

[3]에서 제안한 기법과 본 논문에서 제안한 기법의 연산횟수를 표 3에서 비교하였다. 이때, 3차원 객체는 200×200, 홀로그램은 1,024×1,024 크기로 가정하였다. 표 3을 보면 전연산, 부분승산, 그리고 부분가산의 회수가 각각 99.9%, 99.8%, 66.7% 감소하는 것을 알 수 있다. 여기서 감소율은 가산, 감산, 그리고 승산 연산자들의 사용 횟수만으로 계산된 값이다. 따라서 CGH 연산 속도와 직접적으로 비례 관계에 있는 값은 아니다.

그림 7에서는 본 논문에서 제안한 CGH 알고리듬과 이전 기법들의 CPU 시간의 측정 결과를 보이고 있다. 3차원 객체의 점들의 수를 증가시키면서 제안한 알고리듬과 이전 기법들의 CPU 시간을 측정하여 비교하였다. 그림 6에서 확인할 수 있듯이

표 2. 실험환경
Table 2. Experimental setup

Items		Specification
PC set-up	OS	MS Windows XP
	CPU	Intel Core2 Quad 2.4GHz
	RAM	2.0GB
	Compiler	MS Visual C++ 6.0
Optical parame-ters	3D object size (*xj, yj, zj*)	200×200×8-bit
	Hologram reso-lution (*xa, ya*)	1024×1024
	Reference wave length (λ)	633nm (red)
	Reconstruction distance	1000mm
	Pixel pitch (*p*)	10.4um×10.4um

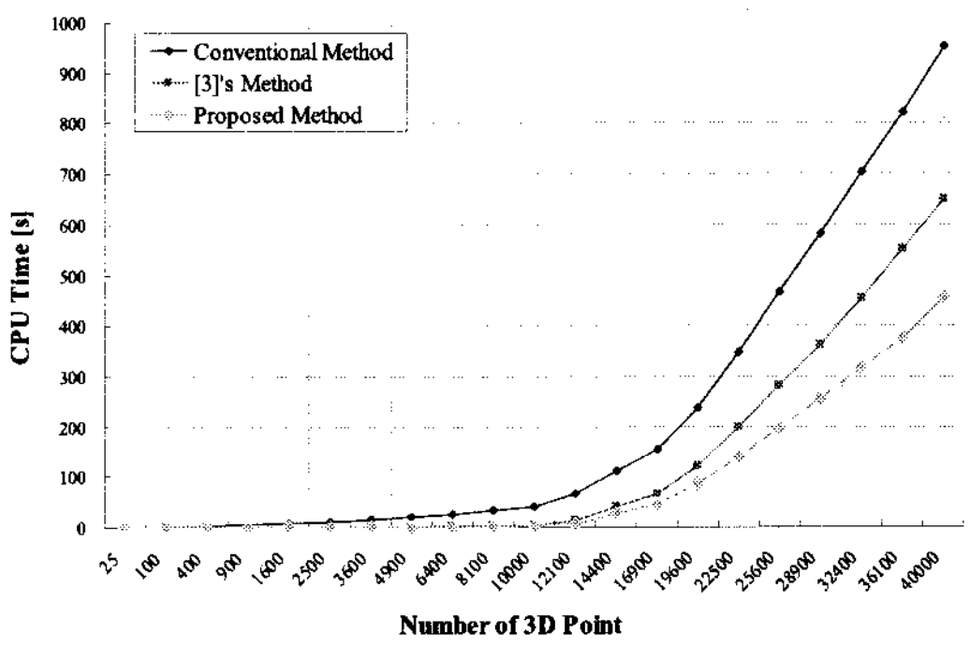


그림 7. 시뮬레이션 결과
Fig. 7. Simulation result

제안한 홀로그램 전체 영역 반복가산 알고리듬은 이전 기법들과 비교하여 최대 30% 이상 연산속도가 빨라졌다.

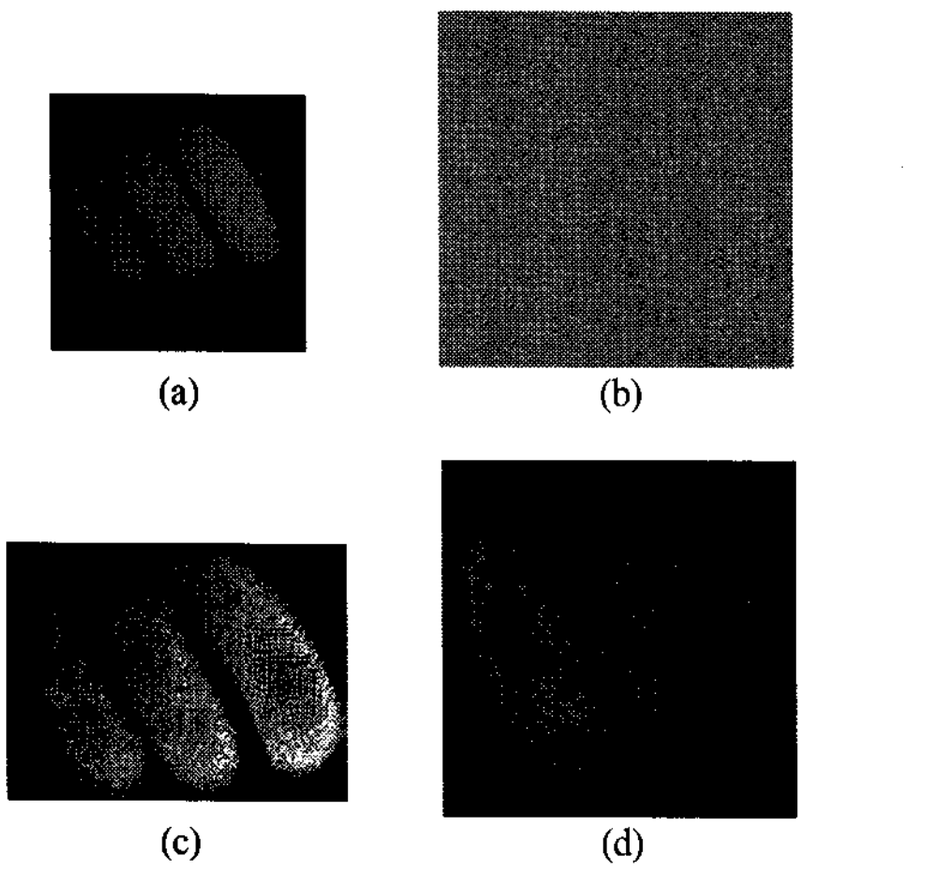
그림 8에서는 실험에 사용한 영상들의 예를 보이고 있다. 그림 8 (a)는 3차원 객체 영상, (b)는 제안한 기법으로 생성한 디지털 홀로그램, (c)는 PC 시뮬레이션으로 복원한 홀로그래피 영상, 그리고 (d)는 광학 시스템에서 복원한 홀로그래피 영상이다. 광학 시스템은 13.62*um*×13.62*um* 화소 간격의 1,280×1,024 크기 SLM을 이용하였고, 532*nm* 파장의 광원으로 복원하였다.

Ⅴ. 결 론

본 논문에서는 실제 객체나 가상의 객체를 디지털 홀로그램으로 변환할 때 사용하는 CGH 알고리

표 3. 연산횟수($\times 10^3$)
Table 3. The number of times($\times 10^3$)

Operation	The number of times		Reduction ratio (%)
	[3]'s	Our's	
Whole operation	1,024×200×200=40,960	1×200×200=40	99.9
Partial multiplication	Γ_{x1}: 1,024×200×200×2=81,920	Γ_{x1}: 1,024×200×200×2=81,920	99.8
	$2\Delta(n-1)$: 1,022×1,024×200×200=41,861,120	Γ_{y1}: 1×200×200×2=80	
Partial addition	Γ_{x1}: 1,024×200×200=40,960	Γ_{xN}: 1,022×1,024×200×200=41,861,120	66.6
	$2\Delta_{x(n-1)}$: 1,022×1,024×200×200×2=83,722,240		
	$\Gamma_{x(n-1)}$: 1,022×1,024×200×200=41,861,120	Γ_{yN}: 1,022×200×200=40,880	



(a) (b) (c) (d)

그림 8. (a) 3차원 객체, (b) 제안한 기법으로 생성한 디지털 홀로그램, (c) (b)를 PC 시뮬레이션으로 복원한 영상, (d) (b)를 광학 시스템에서 복원한 영상
Fig. 8. (a) 3D object, (b) generated digital hologram by proposed method, (c) reconstructed image using PC, (d) reconstructed image using optical system

듬의 연산속도를 높일 수 있는 알고리듬을 제안하였다.

제안한 알고리듬은 3차원 객체의 한 점을 대상으로 디지털 홀로그램의 전체 좌표에서 수행되던 CGH 연산을 첫 번째 좌표에서만 수행하고 나머지 좌표에서는 첫 번째 좌표에서 연산된 결과 값에 이전에 계산된 값들을 반복해서 더해줌으로써 전체 연산시간을 줄일 수 있는 기법이다. 제안한 알고리듬을 이전 기법과 비교한 결과 약 30% 정도 연산속도가 빨라졌다.

본 논문에서 제안한 홀로그램 전체 좌표 반복가산 CGH알고리듬은 추후 차세대 TV로 주목 받는 홀로그래픽 3DTV 시스템의 핵심적인 기반기술이 될 것으로 생각된다.

참 고 문 헌

〔1〕 B. R. Brown and Adolf W. Lohmann, "Complex spatial filtering with binary masks," Applied Optics, Vol.5, pp.967-969, June 1966.

〔2〕 H. Yoshikawa, "Fast computation of Fresnel holograms employing difference," Optical Review, Vol.8, No.5, pp.331-335, 2000.

〔3〕 T. Shimobaba and T. Ito, "An efficient computational method suitable for hardware of computer-generated hologram with phase computation by addition," Computer Physics Communications,vol.138,no.1,pp.44-52,July2001.

〔4〕 M. Lucente, "Interactive Computation of Holograms Using a Look-up Table," Journal of Electronic Imaging, Vol.2, No.1, pp.28-34, Jan. 1993.

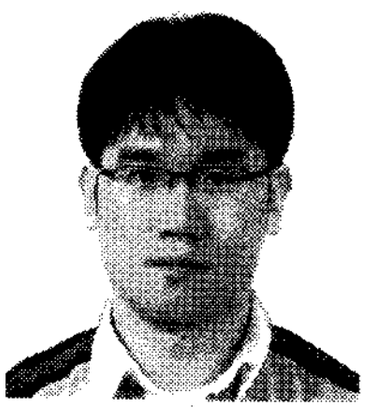
〔5〕 D. E. Smalley, Q. Y. J. Smithwick, and V. M. Bove, Jr., "Holographic video display based on guided-wave acousto-optic devices," Proceedings of SPIEP practical HolographyXXI, Vol.6488, Feb. 2007.

〔6〕 T. Shimobaba, S. Hishinuma, and T. Ito, "Special-purpose computer for holography HORN-4 with recurrence algorithm," Computer Physics Communications, Vol.148, No.2, pp.160-170, Oct. 2002.

〔7〕 T. Ito and T. Shimobaba, "One-unit system for eletroholography by use of a special-purpose computational chip with a high-resolution liquid-crystal display toward a

three-dimensional television," Optics Express, vol.12, no.9, pp.1788-1793,May2004.
〔8〕 T. Ito, N. Masuda, K. Yoshimura, A. Shiraki, T. Shimobaba, and T. Sugie, "Special-purpose computer HORN-5 for a real-time electroholography," Optics Express, Vol.13, No.6, pp.1923-1932, March 2005.
〔9〕 N. Masuda, T. Ito, T. Tanaka, A. Shiraki, and T. Sugie, "Computer generated holography using a graphics processing unit," Optics Express, Vol.14, No.2, pp.603-608, Jan. 2006.
〔10〕 M. Reicherter, S. Zwick, T. Haist, C. Kohler, H. Tiziani, and W. Osten, "Fast digital hologram generation and adaptive force measurement in liquid-crystal-display-based holographic tweezers," Applied Optics, Vol.45, No.5, pp.888-896, Feb. 2006.
〔11〕 T. Haist, M. Reicherter, M. Wu, and,"Using Graphics Boards to Compute Holograms," IEEE Technology Reviews, Vol.8, No.1, pp.8-13, Jan./Feb. 2006.
〔12〕 손정영, 홀로그래피의 원리와 응용, 봉명, 2004.

최 현 준 (Hyun-Jun Choi) 정회원



2003년 2월 광운대학교 전자재료공학과(공학사)
2005년 2월 광운대학교 대학원 졸업(공학석사)
2005년 3월~현재 광운대학교 전자재료공학과 박사과정
<관심분야> Image Processing, 암호학, FPGA/ASIC 설계

서 영 호 (Young-Ho Seo) 종신회원

1999년 2월 광운대학교 전자재료공학과(공학사)
2001년 2월 광운대학교 대학원 졸업(공학석사)
2000년 3월~2001년 12월 인티스닷컴(주) 연구원
2003년 6월~2004년 6월 한국전기연구원 연구원
2004년 8월 광운대학교 대학원졸업(공학박사)
2004년 9월~2004년 11월 유한대학 겸임교수
2004년 12월~2005년 8월 유한대학 연구교수
2005년 9월~2008년 2월 한성대학교 정보통신공학과 교수
2008년 3월~현재 광운대학교 교양학부 교수
<관심분야> Image Processing/Compression, 워터마킹, 암호학, FPGA/ASIC 설계

김 동 욱 (Dong-Wook Kim) 종신회원

1983년 2월 한양대학교 전자공학과 졸업(공학사)
1985년 2월 한양대학교 대학원 졸업(공학석사)
1991년 9월 Georgia공과대학 전기공학과 졸업(공학박사)
1992년 3월~현재 광운대학교 전자재료공학과 정교수. 광운대학교 신기술 연구소 연구원
2000년 3월~2001년 12월 인티스닷컴(주) 연구원
<관심분야> 디지털 VLSI Testability, VLSI CAD, DSP 설계, Wireless Communication