

# 광학 흐름 추정을 위한 새로운 비선형 방법

정희원 김 종 대\*, 박 찬 영\*, 이 용 업\*

## A new nonlinear method for optical flow estimation

Jong Dae Kim\*, Chanyoung Park\*, Yong Up Lee\* *Regular Members*

### 요 약

이 논문은 비선형 운동 제한식을 이용하여 광학 흐름을 반복적으로 구하는 계산식을 제안한다. 선형 운동 제한식을 사용하는 방법 중에 대표적인 방법과 기존의 비선형 운동 제한식을 사용하는 방법을 제안한 방법과 이론적으로 비교 분석하고, 인공 영상 및 실제 영상을 이용한 연속 영상으로 실험하여 제안된 방법이 기존의 방법들보다 성능이 우수함을 보인다.

### ABSTRACT

This paper introduces an iterative solution for the optical flow estimation using the nonlinear motion constraint. The proposed method is analyzed theoretically and compared with the existing methods using the linear and the nonlinear motion constraint. It is proven to have the best performance by testing them for the simulation image sequences that generated from the artificial and the real images.

### I. 서 론

연속하는 영상에서 화소 단위의 움직임을 나타내는 광학흐름(optical flow)은 영상내의 물체의 3차원 구조나 3차원 움직임 등의 동영상 해석을 위한 기본 입력으로 사용할 수 있다<sup>[1]</sup>. 또한 최근에 부상하고 있는 콘텐츠 중심의 저장 및 검색을 위한 움직이는 물체의 분리 등에 중요한 역할을 한다<sup>[2]</sup>. 한편 현재 표준 압축 방식으로 사용되는 블록 단위의 움직임 추정은 실제 광학 흐름을 잘 표현 할 수 없고 복원 시 생기는 블록 왜곡(block artifact) 등의 문제 때문에 광학흐름을 이용하려는 연구들도 발표되고 있다<sup>[3,4,5]</sup>.

광학흐름을 추정하는 방법 중에 대표적인 것으로 특징을 이용하는 방법과 시공간 경사를 이용하는 방법이 있는데, 후자는 다음 식과 같이 영상 내의 밝기 분포를 변경하지 않고 프레임(frame) 간에 이동을 한다는 가정을 한다<sup>[6]</sup>.

$$f(x, y, t) - f(x - u, y - v, t - 1) = 0 \quad (1)$$

여기서,  $f(x, y, t)$ 는 영상 내의 밝기 함수이며, 시간과 공간의 함수로 프레임 두개를 사용하여 얻는다. 식(1)의 첫번째 항은 현재 프레임이고, 두 번째 항은 과거 프레임의 밝기 함수가  $(u, v)$  만큼의 공간 이동한 것을 말한다. 프레임들 사이에 움직임이 작다는 가정 하에서 식(1)에 Taylor 급수 전개를 하여 고차항들을 무시하면, 다음과 같은 선형 운동 제한식을 얻는다.

$$f_x u + f_y v + f_t = 0 \quad (2)$$

여기서  $f_x$ 와  $f_y$ 는 각각 수평, 수직 방향의 공간 경사이고,  $f_t$ 는 시간 경사이다. 속도 성분을 구하기 위하여 또 하나의 제한식이 필요한데, 속도가 공간적으로 크게 변하지 않는다는 평활 제한식(smoothness constraint)을 사용하면 된다. 전술한 두 개 제한식을 이용하여 다음과 같은 최적화 수식

\* 한림대학교 정보통신공학부

논문번호: 00491-1229, 접수일자: 2000년 12월 29일

\* "이 논문은 2001년도 한림대학교 학술연구조성비에 의하여 연구되었음"

을 최소화하는 속도 장을 구하면, 광학흐름을 추정할 수 있다.

$$\iint d(u, v)^2 + \alpha^2 (|\nabla u|^2 + |\nabla v|^2) dx dy \quad (3)$$

여기서  $d(u, v)$ 는 운동 제한식으로 식(1) 또는 (2)의 좌변으로 등가가 되며,  $\alpha$ 는 평활 상수로 운동 제한식과 평활 제한식의 가중치로 볼 수 있다.  $d(u, v)$ 가 선형 제한식일 경우의 최적화 수식(3)의 해를 구하는 문제는 다음 절에 기술한 것처럼 선형 연립 방정식 문제와 같다. 마찬가지로 비선형 운동 제한식(1)을 대입할 경우 비선형 연립 방정식이 도출된다. 그러므로 지금부터 선형 제한식을 사용하여 유도된 해법들을 선형 방법, 비선형 제한식을 사용하는 경우는 비선형 방법으로 한다.

비선형 방법이 선형 방법보다 우수한 결과를 보인다는 연구 결과들이 있다<sup>6,7)</sup>. 이것은 참고문헌[8]에서 분석한 것처럼 선형 운동 제한식을 얻을 때 무시한 고차항 성분들 때문에 최적화 수식(3)의 해에 오차가 생기기 때문이다. 특히 선형 방법은 작은 평활 상수에서 성능이 좋지 않기 때문에 영상 압축 등 운동 제한식에 비중을 두는 응용에서는 바람직하지 않다<sup>5,7)</sup>. 한편 선형 방법으로는 Gauss-Seidel (GS), successive over relaxation(SOR), local relaxation(LR) 등의 여러 가지 반복 계산법이 제안되었으나, 비선형 방법으로는 참고문헌[6]에서와 같이 프레임 보간을 이용하여 점근적으로 선형화 하는 해법 외에는 거의 제안되지 않았다.

본 논문에서는 비선형 방법으로 광학흐름을 구하기 위한 새로운 반복적 계산식을 제안하고 기존의 선형 방법, 비선형 방법과 비교 한다. 2장에서 기존의 선형 방법과 비선형 방법을 설명하고, 3장에서는 새로운 비선형 방법을 도입하고 기존의 비선형 방법과 이론적으로 비교 검토한다. 4장에는 인공 영상과 실제 영상에, 인공 움직임을 적용한 연속영상을 실험하여 결과를 보이고 5장에서 결론을 맺는다.

## II. 기존 방법들의 고찰

### 2.1 기존의 선형 방법

식(3)을 최소화 하기 위하여 Euler-Lagrange 방법을 적용하면 다음과 같은 연립 편미분 방정식을 얻는다.

$$\begin{aligned} \nabla^2 u &= \frac{1}{\alpha^2} f_x d(u, v), \\ \nabla^2 v &= \frac{1}{\alpha^2} f_y d(u, v). \end{aligned} \quad (4)$$

영상을 표본화 하여 이산 공간에서 주사스캔 (raster scan)방식으로 표현하면(lexico-graphical ordering),  $i$  번째 화소의 Laplacian은 5개의 주변 화소로 다음과 같이 근사될 수 있다<sup>9)</sup>.

$$\nabla^2 u_i = \bar{u}_i - u_i = \frac{1}{4} (u_{i-L} + u_{i+L} + u_{i-1} + u_{i+1}) - u_i \quad (5)$$

여기서,  $L$ 은 수평 방향 화소 수이고  $\bar{u}_i$ 는 이웃하는 화소의 수평 방향 속도 성분의 평균이다. 수직 방향 성분도 마찬가지로 표현될 수 있다. 식(5)와 (4)로부터 다음과 같은 연립방정식을 얻을 수 있다.

$$\begin{aligned} \bar{u}_i - u_i &= \frac{1}{\alpha^2} f_x d_i(u_i, v_i), \\ \bar{v}_i - v_i &= \frac{1}{\alpha^2} f_y d_i(u_i, v_i). \end{aligned} \quad (6)$$

선형 운동 제한식으로  $d_i(u_i, v_i)$ 를 대치 하면 식 (6)은 다음과 같은 선형시스템으로 표현된다.

$$\begin{aligned} Ax = b, \end{aligned}$$
$$\begin{aligned} x &= ((u_1, v_1)^T, \dots, (u_N, v_N)^T)^T, \\ b &= -\frac{1}{\alpha^2} (f_x f_x, f_x f_y, \dots, f_x f_x, f_x f_y, f_y f_x, f_y f_y)^T. \end{aligned} \quad (7)$$

화소 수가  $N$ 이고, 행의 화소 수가  $H$ 이면 행렬  $A$ 의 크기는  $2 \times 2$  블록 단위로  $N^2 \times N^2$  블록이며, 대각선 블록(diagonal block)에서 양쪽으로  $H$ 블록 떨어진 곳에  $2 \times 2$  단위 행렬  $I$ 가 존재하는 대형 분산 행렬(large sparse matrix)이 된다<sup>9)</sup>.

식(7)은 여러 반복적 계산법으로 풀어질 수 있는데, 대표적인 것은 앞에서 말한 GS, SOR, LR등이다. 그러나 SOR로 다른 것들을 설명할 수 있으며

로 본 논문에서는 SOR만 적용해 보자. 행렬 A가 2x2 블록 대각선 형태를 가지므로 SOR의 블록형(block-version)을 적용하면, 다음과 같은 반복적 계산식을 얻는다<sup>9)</sup>.

$$\begin{pmatrix} u_i \\ v_i \end{pmatrix}^{n+1} = \begin{pmatrix} u_i \\ v_i \end{pmatrix}^n + \omega D_i^{-1} \left\{ b_i - \sum_{j=1}^{i-1} A^{ij} \begin{pmatrix} u_i \\ v_i \end{pmatrix}^{n+1} - \sum_{j=i}^N A^{ij} \begin{pmatrix} u_i \\ v_i \end{pmatrix}^n \right\}. \quad (8)$$

여기서  $(\cdot)^n$ 은 n번째 반복된 결과 값이며, 식(8)의 성분별 반복 계산식은 다음과 같이 표현된다. 편의를 위해 화소 표시지(pixel index)는 생략한다.

$$u^{n+1} = (1-\omega)u^n + \omega \left\{ \bar{u} - \frac{f_x(f_x \bar{u} + f_y \bar{v} + f_i)}{\alpha^2 + f_x^2 + f_y^2} \right\},$$

$$v^{n+1} = (1-\omega)v^n + \omega \left\{ \bar{v} - \frac{f_y(f_x \bar{u} + f_y \bar{v} + f_i)}{\alpha^2 + f_x^2 + f_y^2} \right\}, \quad (9)$$

$1 < \omega < 2$ .

식(9)에서  $\omega$ 는 이완 인자(relaxation factor)이고,  $\bar{u}$  또는  $\bar{v}$ 는 n번째 단계에서 계산된 주변 화소들의 속도 평균이다. 만일 화소를 주사스캔 방식이 아닌 체크보드(Check Board) 순서로 배열하여 계산하면 반복 계산 한 단계마다 전체 화면을 두 번 스캔(scan)하게 되는데, 두 번째 스캔시의 속도 평균 값은 첫 번째 스캔에서 계산한 속도 값을 평균하면 된다<sup>9)</sup>.

식(9)에서 이완인자  $\omega$ 가 1 일 경우는 GS가 되고 SOR에서는 1부터 2사이의 값을 가지게 된다. 한편 전술한 두 방법 모두 화소 전체에서 같은 값의 이완 인자를 사용하지만 LR에서처럼 지역마다 다른 값을 적용하여 성능을 향상시킬 수 있다<sup>7,10)</sup>.

## 2.2 기존의 비선형 방법

Enkelmann은 광학흐름을 추정하기 위하여 비선형 운동 제한식을 처음으로 적용하였다<sup>6)</sup>. 식(4)에 비선형 운동 제한식을 적용하면 비선형 편미분 방정식이 되지만, 구하고자 하는 속도의 근사치를 알면 선형화시킬 수 있다. 화소에서 속도장의 근사치가  $(u_0, v_0)$ 이고 잔여 속도가  $(u_\delta, v_\delta)$  라면 식(4)은 다음과 같이 근사될 수 있다.

$$\nabla^2 u_\delta = \frac{1}{\alpha^2} f_x \{ u_\delta f_x + v_\delta f_y + d(u_0, v_0) \} + \nabla^2 u_0,$$

$$\nabla^2 v_\delta = \frac{1}{\alpha^2} f_y \{ u_\delta f_x + v_\delta f_y + d(u_0, v_0) \} + \nabla^2 v_0. \quad (10)$$

참고문헌[6]에서와 같이 잔여 속도의 Laplacian을 무시하면 식(10)의 해는 다음과 같다.

$$u_\delta = - \frac{f_x d(u_0, v_0) + (\alpha^2 + f_x^2) \nabla^2 u_0 - f_x f_y \nabla^2 v_0}{\alpha^2 + f_x^2 + f_y^2},$$

$$v_\delta = - \frac{f_y d(u_0, v_0) + (\alpha^2 + f_y^2) \nabla^2 v_0 - f_x f_y \nabla^2 u_0}{\alpha^2 + f_x^2 + f_y^2}. \quad (11)$$

식(11)로부터 얻은 잔여 속도는 근사치  $(u_0, v_0)$ 에 더하여 새로운 근사치로 사용되고, 다시 잔여 속도를 구하는 선형 방정식(10)을 구할 수 있다. 기존의 비선형 방법은 이와 같이 근사 속도를 가지고 전 프레임을 운동 보상하여 보상된 프레임과 현재 프레임에서 선형 방정식을 구하여 잔여 속도를 구하므로 반복적 프레임 보간(successive frame registration, SFR)이라고 명명하기로 한다.

## III. 새로운 비선형 방법의 제안과 기존의 비선형 방법과의 비교

### 3.1 새로운 비선형 방법

식(6)의 편 미분 방정식에 식(1)을 적용하면 다음과 같은 비선형 편 미분 방정식을 얻는다.

$$\bar{u}_i - u_i = \frac{1}{\alpha^2} f_x (f(x_i, y_i, t) - f(x_i - u_i, y_i - v_i, t - 1)),$$

$$\bar{v}_i - v_i = \frac{1}{\alpha^2} f_y (f(x_i, y_i, t) - f(x_i - u_i, y_i - v_i, t - 1)) \quad (12)$$

여기서, 주변 화소의 편 미분 방정식은 서로 변수를 공유하기 때문에 전체 화면에 대한 비선형 시스템을 새로 정의해야 한다. 전체 비선형 시스템을 표현하기 위하여 다음 벡터들을 정의 하자.

$$z_i = (u_i, v_i)',$$

$$z = (z_1, \dots, z_i, \dots, z_N)',$$

$$I_i^u(z) = u_i - \bar{u}_i + \frac{1}{\alpha^2} f_x (f(x_i, y_i, t) - f(x_i - u_i, y_i - v_i, t - 1)),$$

$$I_i^v(z) = v_i - \bar{v}_i + \frac{1}{\alpha^2} f_y (f(x_i, y_i, t) - f(x_i - u_i, y_i - v_i, t - 1)),$$

$$I_i = \begin{pmatrix} I_i^u(z) \\ I_i^v(z) \end{pmatrix} = z_i - \bar{z}_i + \frac{1}{\alpha^2} \begin{pmatrix} f_x \\ f_y \end{pmatrix} (f(x_i, y_i, t) - f(x_i - u_i, y_i - v_i, t - 1)) \quad (13)$$

그러면, 광학 흐름을 얻는 문제는 다음과 같은 비선형 시스템으로 표현할 수 있다.

$$L(\mathbf{z}) = (\mathbf{l}_1, \dots, \mathbf{l}_i, \dots, \mathbf{l}_N)' = 0. \quad (14)$$

식(14)는 시스템 L의 Jacobian 행렬에서 대각선이나 블록 대각선(block-diagonal)을 이용하여 반복적 계산식을 얻을 수 있다. L의 Jacobian 행렬에서 대각선 블록들로 구성된 행렬을 W라고 하면 다음과 같은 single-step Newton 방법(SNM)을 이용한 반복적 계산법을 얻을 수 있다<sup>[11]</sup>.

$$\mathbf{z}^{n+1} = \mathbf{z}^n - \omega W^{-1} L(\mathbf{z}^n) \quad (15)$$

L의 Jacobian은 식(9)의 A와 같으므로 화소 별로 다음과 같은 반복적 계산식을 얻는다.

$$\mathbf{z}_i^{n+1} = \mathbf{z}_i^n - \omega D_i^{-1} \mathbf{l}_i(\mathbf{z}_i^n) \quad (16)$$

식(16)을 화소표지자를 생략하고 속도 성분별로 정리하면 다음과 같다.

$$\begin{aligned} u^{n+1} &= (1-\omega)u^n + \omega \left\{ \bar{u} - \frac{f_x(f_x \bar{u} + f_y \bar{v} + d(u^n, v^n)) - f_x u^n - f_y v^n}{\alpha^2 + f_x^2 + f_y^2} \right\}, \\ v^{n+1} &= (1-\omega)v^n + \omega \left\{ \bar{v} - \frac{f_y(f_x \bar{u} + f_y \bar{v} + d(u^n, v^n)) - f_x u^n - f_y v^n}{\alpha^2 + f_x^2 + f_y^2} \right\}. \end{aligned} \quad (17)$$

식(17)에 이동 시간 경사(displaced frame difference)  $d(u^n, v^n)$ 의 근사치를 대입하면 보다 간단한 수식을 얻을 수 있다. 먼저  $d(u^n, v^n)$ 를 (0,0)에서 Taylor 전개하여 1차식만을 취하면 식(9)와 같아서 선형 운동 제한식을 사용한 경우와 같은 형태의 반복적 계산식을 얻을 수 있다. 한편 반복적 계산 도중에 해답에 가장 가까운 속도는 바로 전 단계에서 구한 주변속도 평균인  $(\bar{u}^n, \bar{v}^n)$ 이므로 이 속도에서 Taylor전개 하면 다음과 같은 간단한 식을 얻을 수 있다.

$$\begin{aligned} u^{n+1} &= (1-\omega)u^n + \omega \left\{ \bar{u} - \frac{f_x d(\bar{u}^n, \bar{v}^n)}{\alpha^2 + f_x^2 + f_y^2} \right\}, \\ v^{n+1} &= (1-\omega)v^n + \omega \left\{ \bar{v} - \frac{f_y d(\bar{u}^n, \bar{v}^n)}{\alpha^2 + f_x^2 + f_y^2} \right\}. \end{aligned} \quad (18)$$

식(18)은 참고문헌[7]에서 직관적으로 구한 식과 일치한다. 식(18)과 (17)은 그 성능이 거의 차이가 없어서 실험 시에 식(17) 대신에 식(18)을 사용하기로 한다.

### 3.2 기존의 비선형 방법과의 비교

2.2절에서 설명한 기존의 비선형 방법은 근사치로 문제를 선형화 하여 풀고 그 해답을 다시 근사치로 사용 하여 다시 문제를 선형화 하는 방법이다. 근사치가  $(u_0, v_0)$ 이면 선형화 한 후의 잔여 속도에 대한 선형 방정식은 식(11)과 같고 이 수식에 SOR을 적용하면 다음과 같은 반복적 제한식을 얻을 수 있다.

$$\begin{aligned} u_\delta^{k+1} &= (1-\omega)u_\delta^k + \omega \left\{ \bar{u}_\delta - \frac{f_x(f_x \bar{u}_\delta + f_y \bar{v}_\delta + d(u_0, v_0)) + s_x(u_0, v_0)}{\alpha^2 + f_x^2 + f_y^2} \right\}, \\ v_\delta^{k+1} &= (1-\omega)v_\delta^k + \omega \left\{ \bar{v}_\delta - \frac{f_y(f_x \bar{u}_\delta + f_y \bar{v}_\delta + d(u_0, v_0)) + s_y(u_0, v_0)}{\alpha^2 + f_x^2 + f_y^2} \right\}, \\ s_x(u_0, v_0) &= (\alpha^2 + f_y^2)(\bar{u}_0 - u_0) - f_x f_y (\bar{v}_0 - v_0), \\ s_y(u_0, v_0) &= (\alpha^2 + f_x^2)(\bar{v}_0 - v_0) - f_x f_y (\bar{u}_0 - u_0). \end{aligned} \quad (19)$$

식(19)에서  $\omega$ 를 1로 하고, 한번만 반복 계산을 한다면 식(11)과 같아져서 2.2절에서 설명한 기존의 비선형 방법과 같아지게 된다. 그러나 체크보드 순서로 계산하면 1번째 스캔에서 계산된 값을 2번째 스캔에서 사용할 수 있어서 제안한 비선형 방법은 한번만 반복 계산해도 기존의 비선형 방법보다 정확한 해를 얻을 수 있다. 또한 SOR이나 LR에서는  $\omega$ 가 1보다 큰 값을 사용할 수 있으므로 수렴 속도가 빠를 뿐이 아니라, 실제로는 본 논문의 실험 결과에서 보듯이 더 정확한 값을 얻을 수 있다. 식(19)에서  $(u_0, v_0)$ 를 전체 반복식의 전 단계(step)에서 구한 값  $(u^n, v^n)$ 을 사용하고 한번의 반복으로 잔여 속도를 구하면 다음과 같은 식들에 의하여 식(19)는 식(17)과 같아진다.

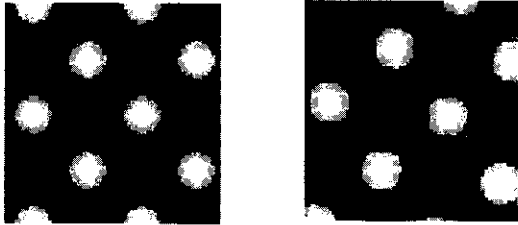
$$\begin{aligned} (u_\delta^0, v_\delta^0) &= 0, \\ (u^n, v^n) &= (u_0, v_0), \\ (\bar{u}^n, \bar{v}^n) &= (\bar{u}_0 + \bar{u}_\delta, \bar{v}_0 + \bar{v}_\delta), \\ (u^{n+1}, v^{n+1}) &= (u^n, v^n) + (u_\delta^1, v_\delta^1). \end{aligned} \quad (20)$$

그러므로 제안한 비선형 방법으로 기존의 비선형 방법보다 빠르고 정확하게 해를 구할 수 있다. 본 논문에서는 수렴 속도보다 중요한 정상 상태에서의 오차(steady state error)를 비교하여 제안한 방법의 우수함을 보이고자 한다.

## IV. 실험 및 결과

여러 가지 반복적 계산법은 평활 상수 값에 따라 다른 특성을 나타내기 때문에 고정된 값에 비교하

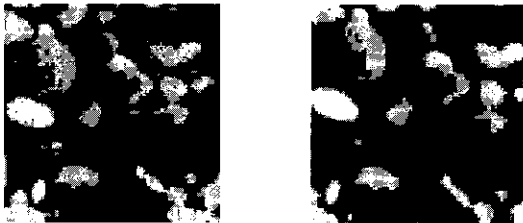
는 것은 의미가 없다. 본 논문에서는 평활 상수의 변화에 따른 정상 상태 오차를 비교하여 그 성능을 비교하였다.



(a) 과거 프레임

(b) 현재 프레임

그림 1. 인공 영상으로 만든 프레임 쌍



(a) 과거 프레임

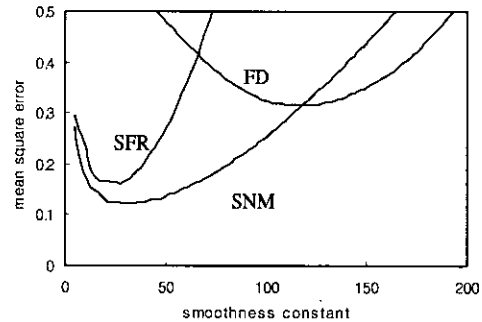
(b) 현재 프레임

그림 2. 실제 영상으로 만든 프레임 쌍

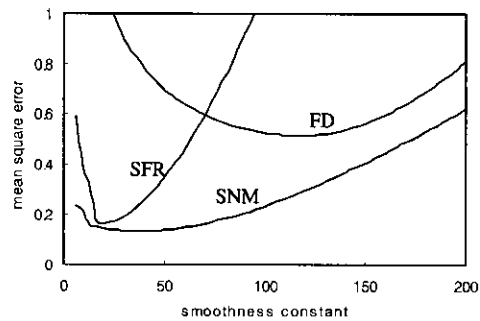
제안한 비선형 방법, 기존의 선형 방법 그리고 기존의 비선형 방법을 각각 SNM, FD 그리고 SFR로 표기 하였다. 성능의 비교를 위하여 실제 상황을 대신할 수 있는 여러 가지 인공 상황을 만들었다. 그림 1에는 밝기가 수직 정현파와 수평 정현파의 곱으로 표현되는 영상과 그 영상을 (1,0) 화소만큼 이동, 6° 회전, 1.1배 확대한 영상으로 다음 프레임을 만들었다(그림 1(b)). 그림 2(a)는 실제 영상에서 잘라 내 온 것이고, 그 영상을 그림 1(b)를 만들 때 사용한 움직임을 적용하여 그림 2(b)에서 보인 것처럼 다음 프레임을 만들었다.

그림 3은 기존의 선형 방법(FD), 비선형 방법(SFR) 과 제안한 비선형 방법(SNM)을 적용 했을 때의 평활 상수 값의 변화에 따른 정상 상태 오차의 변화를 보인다. 그림에 나타난 바와 같이 선형 방법보다 비선형 방법들이, 인공 영상이나 실제 영상에서 모두 월등한 성능을 보인다. 선형 방법의 경우, 상대적으로 큰 값의 평활 상수에서 가장 좋은 성능을 보이는 것은 선형 제한식 자체의 오차가 크기 때문이다. 한편 기존의 비선형 방법(SFR)은 성능이 상

당히 좋아지는 평활 상수 구간은 있으나 그 폭이 너무 작아서 안정성에 문제가 있다. 실제 연속 영상에서 최적한 평활 상수를 구하기 위하여 여러 가지 방법이 개발되어 있지만 계산된 속도의 오차를 최소화 하는 평활 상수를 구하기는 어렵다. 그러므로 제안한 비선형 방법(SNM)처럼 어느 정도 넓은 구간에서 속도 오차가 작아야 유리할 것이다. 참고적으로 실제 광학흐름과 제안한 방법(SNM)에서 구한 광학흐름을 각각 그림 4(a)와 (b)에 도시하였다.



(a) 그림1 프레임 쌍에 대한 결과 (SNM: 제안한 방법, FD: 기존의 선형 방법, SFR: 기존의 비선형 방법)

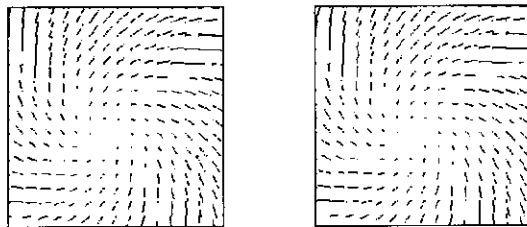


(a) 그림2 프레임 쌍에 대한 결과 (SNM: 제안한 방법, FD: 기존의 선형 방법, SFR: 기존의 비선형 방법)

그림 3. 그림1과 그림2에 보인 프레임 쌍들에 3가지 방법을 적용한 결과

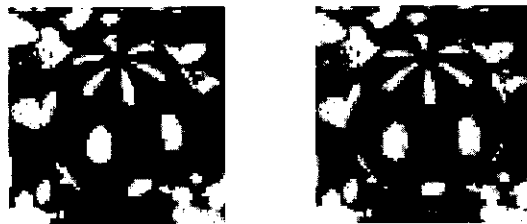
그림 5는 운동 경계가 있는 영상에서 제안한 방법(SNM)과 기존의 방법들(FD, SFR)과의 성능을 비교하기 위하여 만든 인공 영상이다. 그림의 배경은 그림 1(a)에서 보인 실제 영상이고 전경에는 표면 밝기가 수직 정현파와 수평 정현파의 곱으로 표시되는 구가 앞을 향하여 45° 기울어져서 왼쪽으로 회전한다. 그림 5(b)에서의 구는 그림 5(a)에서보다

왼쪽으로  $10^\circ$  더 회전했다. 그림 6에 그 결과를 보인다. 그림 6(a)에서 보듯이 세 가지 방법 모두에서 성능의 차이가 줄어들어 보이는 데 그 이유를 알기 위하여 그림 7에 실제 속도와 세 가지 방법으로 구한 실제 속도들을 보였다. 3가지 방법 모두 운동 경계의 바깥에서 실제 속도는 0이어야 하지만 반복적 계산식의 특성상 0이 아닌 부분에서 구한 속도가 퍼져 나가는 것을 알 수 있다. 운동 경계에 대한 처리는 본 논문에서 다루고자 하는 부분이 아니므로 그림 6(b)에서처럼 구 안에서만 정상 상태 오차를 구하여 비교하는 것은 의미가 있을 것이다. 그림 6(b)에서 보듯이 운동 경계가 없는 영상의 경우에서와 마찬가지로 성능의 차이를 보인다.



(a) 실제 속도 (b) 계산된 속도

그림 4. 그림1의 프레임 쌍에 제안한 방법을 적용하여 구한 속도장(smoothness constant = 30)



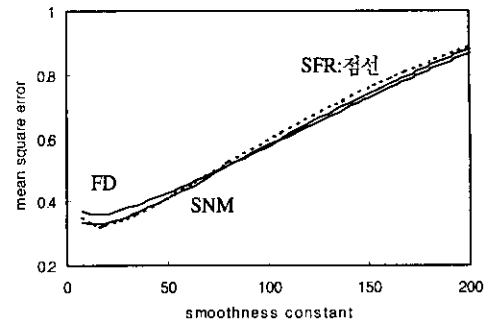
(a) 과거 프레임 (b) 현재 프레임

그림 5. 운동 경계가 있는 영상의 예. 실제 영상을 배경으로 인공 구 영상이 앞으로 기울어져서 왼쪽으로 회전하였다

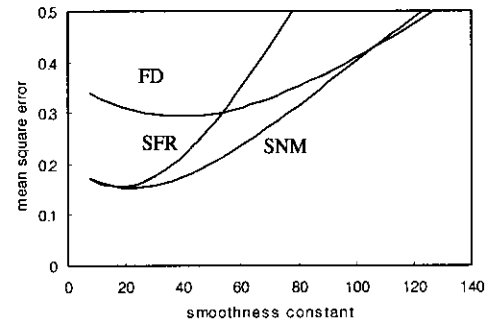
## V. 결론

광학 흐름을 추정하기 위하여 비선형 운동 제한식을 사용하는 비선형 방법의 새로운 반복적 계산식을 제안하고 기존의 선형 방법 및 비선형 방법과 이론적, 실험적으로 비교 검토 하였다.

인공 영상과 실제 영상을 인공 운동으로 만든 연속 영상의 쌍을 가지고 실험하여 제안한 방법의 성



(a) 그림5의 프레임 쌍에 3가지 방법을 적용한 결과 (SNM:제안한 방법, FD:기존의 선형방법, SFR:기존의 비선형 방법)



(b) 그림5의 프레임 쌍에 3가지 방법을 적용하여 구 안에서만 평균 자승 오차를 구하였다. (SNM:제안한 방법, FD:기존의 선형 방법, SFR:기존의 비선형 방법)

그림 6. 그림5의 프레임 쌍에 3가지 방법을 적용한 결과

능이 가장 우수함을 보였다. 운동 경계가 있는 연속 영상에서도 경계 부분에서 예측 속도의 퍼짐 현상은 기존의 방법들과 비슷하나 경계가 아닌 부분에서는 역시 제안한 방법이 성능이 우수하다는 것을 보였다.

제안한 방법의 반복식은 기존의 비선형 방법과는 달리 기존의 선형 방법과 같은 단일 구조로 되어 있으므로 광학흐름을 예측하는 여러 가지 추가적인 기법에 사용 될 수 있다. 가령 다층 구조(hierarchical)로 광학흐름을 예측 할 때는 상위 층(higher level)에서 얻은 예측 속도를 초기 속도장으로 사용하여 반복식을 적용하면 된다. 시간적으로 반복하여 광학흐름을 구할 때도 마찬가지로 과거 프레임까지 예측한 광학 흐름을 초기 값으로 적용할 수 있다.

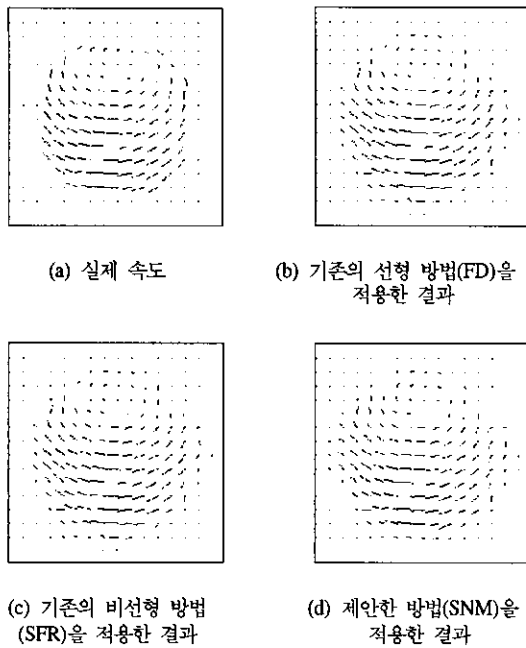


그림 7. 그림5의 프레임 쌍에 각 방법으로 구한 속도장 (smoothness constant=2)

참 고 문 헌

[1] G. Adiv, "Determining 3-d motion and structure from optical flow generated by several moving objects," *IEEE tr. on PAMI*, Vol. 7, No. 4, pp. 384-401, July 1985.

[2] S. Nitsuwat, J. S. Jin and H. M. Hudson, "Motion-based video segmentation using fuzzy clustering and classical mixture model," in *Proc. of IEEE Inter. Conf. On Image Processing*, Vol. 1, pp. 300-303, IEEE press, Sept. 2000.

[3] S. Han and C. Podilchuk, "Modeling and coding of dfd using dense motion fields in video compression," in *Proc. of IEEE Inter. Conf. On Image Processing*, IEEE press, Vol. 3, pp. 146-149, Sept. 2000.

[4] E. Memin, P. Perez, "Dense estimation and object-based segmentation of the optical flow with robust techniques," *IEEE tr. on Image Processing*, Vol. 7, No. 5, pp. 703-719, May 1998.

[5] J. D. Kim, "Motion compensation of wavelets with implicit optical flow," *Proceedings of*

*SPIE*, Vol. 3024, pp. 1360-1369, 1997.

[6] W. Enkelmann, "Investigations of multigrid algorithms for the estimation of optical flow fields in image sequences," *Computer Vision, Graphics, and Image Processing*, Vol. 43, pp. 150~177, 1988.

[7] J. D. Kim and S. K. Mitra, "A local relaxation method for optical flow estimation," *Signal Processing: Image Communication*, Vol.11, No.1, pp.21~38, 1997.

[8] T. S. Denny and J. L. Prince, "Optimal brightness functions for optical flow estimation of deformable motion," *IEEE tr. on Image Processing*, Vol. 3, No. 2, pp. 178-191, March 1994.[9] W. Hackbusch, *Iterative Solution of Large Sparse Systems of Equations*, New York, NY: Springer-Verlag, 1995.

[10] C.-C. J. Kuo, B. C. Levy, and B. R. Musicus, "A local relaxation method for solving elliptic PDEs on mesh-connected arrays," *SIAM Journal on Scientific and Statistical Computing*, Vol.8, pp.550~573, 1987.

[11] U. M. Theodor Meis, *Numerical solution of partial differential equations*, New York: Springer-Verlag, 1981.

김 종 대(Jong-dae Kim)

정희원



1982년 2월 : 서울대학교

전자공학과 (공학사)

1984년 2월 : 한국과학기술원

전기및 전자공학과  
(공학석사)

1990년 2월 : 한국과학기술원

전기및 전자공학과  
(공학박사)

1988년 3월~2000년 2월 : 삼성전자주식회사 중앙연구  
소 수석연구원

1994년 12월~1995년 12월 : UCSB 방문연구자

1996년 1월~1997년 2월 : UCLA 방문연구자

2000년 3월~현재 : 한림대학교 정보통신공학부 조교  
수

<주관심 분야> 신호 및 영상처리, 멀티미디어 통신

