

자동초점조절 기법및 하드웨어의 구현

正會員 張 成 翼*, 李 龍 萬, 鄭 起 旭*, 金 元 燦*, 李 忠 雄*

Auto-Focusing Technique and Its Hardware Implementation

Seong Ik Jang*, Yong Man Lee, Ki Wook Chung*, Won Chan Kim*, Choong Woong Lee* Regular Members

要 約

화상의 질을 떨어뜨리는 요인들 중 가장 중요한 것이 초점이 정확히 맞지 않아 발생하는 화상의 흐림 현상이다. 화상의 흐림 현상은 초점을 맞추는데 사용되는 알고리즘이 현재 촬영되는 상황에 적합하지 않은 경우에 발생한다. 일반적으로 화상은 매우 복잡한 형태를 띠므로 통계적으로 간단히 모델링할 수 없다. 더욱이 정지 화상이 아닌 움직이는 피사체를 담고 있는 화상의 경우에는 시간적인 변화를 고려해야 하므로 고정적인 알고리즘으로는 항상 정확한 초점조절을 하는 것이 불가능하다. 따라서 촬영되고있는 화면의 특성을 분석하여 그에 따라 적응적으로 초점조절 방법 및 파라미터를 가변하는 알고리즘이 필수적이다. 본 논문에서는 이와 같이 화면에 적응적으로 반응할 수 있는 자동초점조절 프로세서를 설계하고 알고리즘을 개발하여 실제로 자동초점조절 장치를 제작, 실험하였다. 피사체가 움직이면 초점이 맞은 상태에서도 초점값이 계속 변화하므로 적응형 자동초점조절 장치는 피사체의 움직임 을 고려할 수 있어야 한다. 따라서 제작한 칩은 화면을 64개의 영역으로 분할하여 각 영역에서의 초점값을 따로 구하여 출력하며 영역 단위 정도의 정확도로 움직임을 추출할 수 있는 정보를 제공한다. 또, 상황에 따라 화상의 주파수 성분이 크게 달라지므로 초점값은 고정된 주파수 대역만을 이용하는 것이 아니라 대역을 전환하면서 초점 값을 구할 수 있다. 또한 다양한 상황 판단을 위해 퍼지 보조 연산기를 내장하여 초점값 연산과는 독립적으로 작동된다.

ABSTRACT

The most important factor degrading the images is the blurring effect due to inexact focusing. The blurring effect occurs when the focusing algorithm is not suitable for the situation. Because images are complicated, modelling is impractical. Moreover, moving images require time-varying considerations so that the focusing algorithm should be an adaptive one, that is, one that adapts the

*서울대학교 전자공학과

Department of Electronics Engineering, Seoul
National University

論文番號 : 94282

接受日字 : 1994年 10月 13日

method and parameters to the time-varying features of the image. In this thesis, a scene-adaptive auto-focusing processor and an algorithm are designed and an auto-focusing system with those is implemented and tested. The adaptive system must detect the motion of object in order not to be affected by the fluctuation of focus-value of the moving object. Thus the processor segments the screen into 64 blocks and computes the focus-value of each block and motion information. In addition, the sensitivity of focus-value to focusing states varies with the situations, therefore the processor computes various focus-values. It also has an internal fuzzy coprocessor for decision of various situations with minimal hardware and software.

1. 서론

카메라의 자동초점조절은 컴퓨터 비전과 현미경 및 일반 카메라나 비디오 카메라에서 매우 중요하게 사용된다. 자동초점조절 방식은 크게 능동형(active)과 수동형(passive)의 2가지로 구분된다. 능동형 자동초점조절 방식은 적외선 또는 초음파 등의 측정용 신호를 카메라로부터 발사하고, 피사체에 반사되어 돌아오는 신호를 이용, 거리를 계산하여 초점을 맞추는 방식이다. 수동형 자동초점조절 방식은 카메라 외부로 자동초점조절을 위한 신호를 발사하지 않고 촬영을 위하여 받아들이는 영상 신호 자체를 이용한다는 의미에서 '수동'이라는 이름을 가지며 영상 처리형 방법이라고 불리기도 한다. 수동형 자동초점조절 방식의 구체적인 방법은 다양하지만 휘도 신호로부터 초점조절에 유용한 초점 평가치(이하 초점값)를 얻어 내어 그 값이 최대가 되는 곳으로 렌즈를 구동시키는 것이 일반적이다⁽¹⁾. 그림 1에 그 과정이 나타나 있는데 곡선에 표시된 점에 써 있는 숫자는 시간적인 순

서를 의미한다. 스텝 1에서 시작하여 초점값이 커지는 쪽으로 계속 이동하다가 스텝 7에서 그 값이 감소하면 이전 위치가 최대값이라고 생각하고 이전의 위치로 이동한 다음 정지한다. 만일 처음에 이동했을 때 초점값이 감소했다면 그 반대방향으로 추적을 한다⁽²⁾.

능동형 방식은 정확한 초점 위치를 찾는 것이 어렵고 부가적인 장치가 따라야 하며 피사체의 종류에 따라 반사되는 신호가 영향을 받을 수 있다는 점 등의 단점을 갖고 있기 때문에 현재는 수동형 방식이 주로 사용되고 있다. 수동형 방식에 대해서 현재까지 다수의 연구논문이 발표되었지만 대부분 초점값을 어떻게 정의할 것인가에 집중되어왔다⁽³⁻⁵⁾. 이것은 초기의 자동초점조절 연구의 대상이 대부분 현미경 분야이었던 것에 기인한다. 현미경의 경우에는 피사체가 고정되어 있으므로 합당한 초점값을 정의하기만 하면 그에 따라 렌즈의 위치를 이동시켜가면서 초점값의 최대값을 찾으면 된다. 따라서 초점이 맞는 정도에 따라 단조 증가하며 초점이 정확히 맞은 지점에서 최대

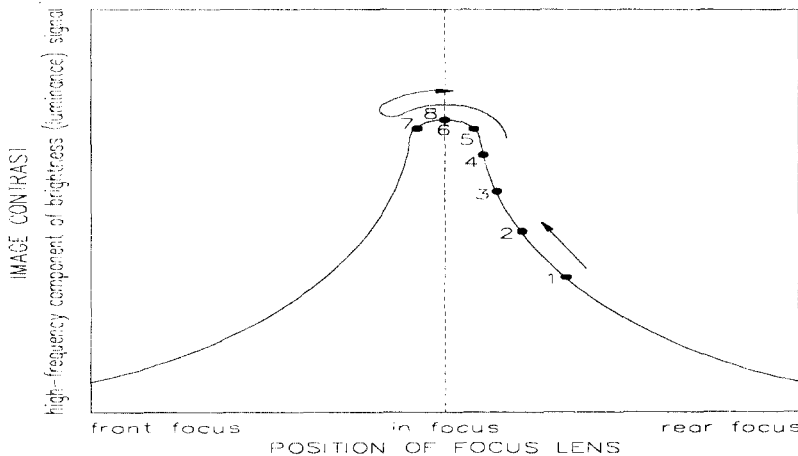


그림 1. 초점값 곡선의 예
Fig. 1. Example of focus value curve

가 되는 초점값을 정의하는 것이 가장 중요하다. 그러나 비디오 카메라와 같은 경우에는 피사체가 정지해 있는 경우보다 움직이고 있는 경우가 훨씬 많다. 또한 일단 초점을 맞추었다고 하더라도 그것으로 끝나는 것이 아니라 피사체가 바뀌거나 카메라를 움직이는 경우 새로이 초점을 맞추어야 하는지 아니면 그 전 상태를 유지해야 하는지도 판단을 해야 한다. 더욱이 실내에서만 촬영하는 것이 아니므로 밝기, 심도, 피사체의 거리, 줌의 정도, 피사체의 수, 피사체와 배경과의 관계 등 촬영 상황이 매우 다양하다. 따라서 단순히 초점값을 잘 정의하는 것보다는 화면 상태의 판단 및 그에 따른 적절한 초점값의 적용이 중요하게 된다. 또한 피사체가 계속적으로 변동하게 되므로 초점조절이 빠르게 이루어져야 한다. 즉 초점값을 구하는 것과 이를 이용한 렌즈위치의 조절이 충분히 빠르게 이루어지지 않으면 이미 바뀌어져 있는 피사체에 대해서 이전의 피사체의 거리로 초점을 맞추는 일이 벌어지게 된다.

지금까지의 연구들은 이러한 점에서 비디오 카메라에 적용하기에는 무리가 있었다. 이론을 검증하기 위한 실험방법에 있어서도 미리 얻어진 순차 영상들에 대하여 컴퓨터 시뮬레이션을 하거나 CCD 카메라를 사용하는 경우에도 Frame Grabber Board를 이용하여 한 프레임을 그래프한 다음 이 데이터를 이용하여 PC 나 워크스테이션등에서 초점값을 구하고 구해진 값에 따라 렌즈를 구동한 후 다시 앞의 과정을 반복하는 방법을 사용하였다. 이러한 방법을 사용하게 되면 초점값을 구하는 간격이 매우 길게 되어 움직이는 피사체의 경우에는 실험을 할 수 없다.

본 논문에서는 이전까지의 연구와는 달리 이러한

점에 주목하여 실시간으로 처리가 가능하면서도 앞서 언급한 성질을 만족시키는 초점값을 정의하고 초점이 맞지 않아 발생하는 흐림과 움직임으로 인한 흐림을 효율적으로 이용하여 기존의 영상처리형 자동초점조절 방법이 가졌던 약점을 보완할 수 있는 알고리즘을 고안하였으며 이를 하드웨어로 구현하여 검증한 실험 결과를 소개한다. 먼저 II장에서는 흐림현상의 모델에 대하여 알아보고, III장에서 하드웨어 구조는 간단하면서도 화상의 특징을 잘 반영하는 초점값을 얻는 방법에 대해 설명하며, IV장에서는 움직임 추정방법과 그 활용에 대하여 설명하였다. 하드웨어 구현은 V장에서 다루어지며 실험 결과 및 검토를 VI장에서 다루고 VII장에서 결론을 맺는다.

II. 흐림현상의 모델

화상은 3 차원의 물체나 풍경이 카메라 등과 같은 기록 매체에 의해서 2차원의 면에 맺히는 것이다. 이러한 화상 형성 과정을 선형으로 가정한다면 화상은 결국 그림 2와 같은 시스템의 출력으로 나타낼 수 있고 이것을 수식으로 나타내면 식 (1)과 같다.

$$g(x, y) = s \left[\int_{-\infty}^{\infty} \int_{-\infty}^{\infty} h(x, y; s, t) \cdot f(x, y) \cdot ds dt \right] + n(x, y) \quad (1)$$

이 때 식 (1)의 $g(x, y)$ 는 기록되는 화상을 나타내고 $f(x, y)$ 는 실제의 3 차원 풍경이 2차원으로 표시된 것을 나타낸다. $h(x, y; s, t)$ 는 선형 흐림 시스템의 2차원 점확산함수(Point Spread Function)이고 s 는 CCD 와 같은 촬상소자가 가지는 기록감도의 비선

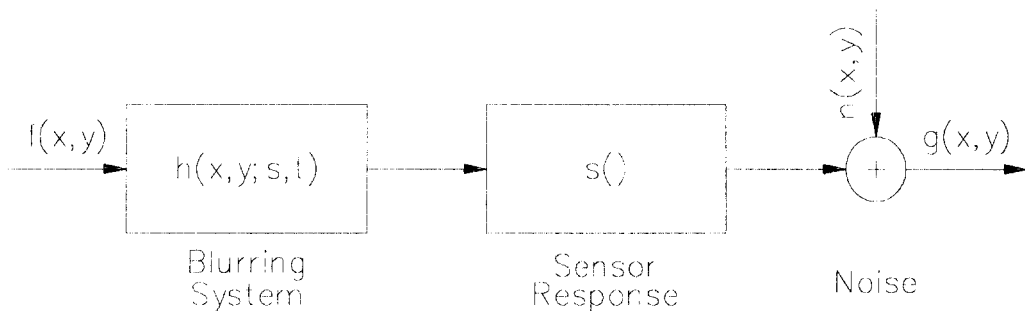


그림 2. 화상의 형성과 기록과정의 모델
Fig. 2. The model of mapping and recording process of image

형 함수이다. 화상의 기록시 발생하는 잡음은 화상과 통계적으로 무관한 가산성 잡음으로 나타내었다. 물론 필름면의 자성체 분포나 광자의 발생 통계에 따른 잡음 등은 화상과 통계적으로 무관하지는 않지만 대부분의 경우 이런 모델로 충분하며 잡음이 없는 경우를 가정하더라도 만족할 만한 결과를 얻을 수 있기 때문에, 본 논문에서는 잡음이 없는 상태를 가정한다.

만일 위의 점확산함수가 화상 전체나 일정한 구간에서 안정적(stationary)이라면 단순히 x-s와 y-t의 함수가 되어 식은 다음과 같은 길쌈적분(convolutional integral)이 된다.

$$g(x, y) = s \left[\int_{-\infty}^{\infty} \int_{-\infty}^{\infty} h(x-s, y-t) \cdot f(s, t) ds dt \right] \quad (2)$$

$$g(x, y) = s [h(x, y) * f(x, y)] \quad (3)$$

이산 신호의 경우에는 연속 변수 f, g, h 등이 모두 등간격 샘플링되면 N×N 행렬로 나타나고 시스템 모델은 다음과 같은 이산 신호의 식으로 표현된다.

$$g(i, j) = s \left[\sum_{\forall(k, l)} h(i, j; k, l) f(k, l) \right] \quad (4)$$

$$0 \leq i, j \leq N-1$$

또한 연속 신호에서와 마찬가지로 시스템이 공간적으로 안정한(spatially invariant) 경우에는 위의 합이 이산 길쌈 합으로 변환된다.

$$g(i, j) = s \left[\sum_{\forall(k, l)} h(i-k, j-l) f(k, l) \right] \quad (5)$$

$$0 \leq i, j \leq N-1$$

$$g(i, j) = s [h(i, j) * f(i, j)] \quad (6)$$

$$0 \leq i, j \leq N-1$$

촬상소자 감도의 비선형성은 분석의 어려움을 가져오므로 때때로 무시되거나 선형으로 가정하게 된다. 만일 이와 같은 가정이 크게 틀리지 않은 시스템인 경우에는 식 (6)은 더욱 간단해져 다음과 같은 선형

길쌈 모델이 된다.

$$g(i, j) = h(i, j) * f(i, j) \quad 0 \leq i, j \leq N-1 \quad (7)$$

이것을 푸리에 변환하게 되면 식 (8)과 같이 원 신호성분의 주파수 신호와 시스템의 주파수 응답의 곱으로 나타낼 수 있다.

$$G(m, n) = H(m, n)F(m, n) \quad (8)$$

H(m, n) : 흐림 시스템의 주파수 응답

F(m, n) : 원 화상신호의 주파수 신호

m : 수평 방향의 공간 주파수

n : 수직 방향의 공간 주파수

자동초점조절을 위해서는 (8)식에서 H(m, n)으로 나타낸 흐림 시스템의 특성을 알아야 한다. 이러한 흐림 현상은 초점이 맞지 않아 생기는 것과 움직임으로 인하여 생기는 것으로 나눌 수 있다.

초점이 맞지 않는 흐림현상은 촬상면이 피사체의 거리에 대하여 적합한 위치에 있지 않을 때 발생한다. 이것은 피사체 표면의 점이 촬상면위의 한 점에 대응되지 않고 확산되기 때문이다. 이 확산은 점 광원이 원으로 투영되는 현상으로 근사할 수 있다. 이 원을 확산원(Circle of Confusion)이라 하며 그림 3은 피사체와 렌즈계 사이의 거리에 따른 확산원 지름의 변화를 보여준다^(3,4). 점광원은 b의 위치에 있을 때 정확히 초점이 맞은 것이며 그 앞이나 뒤에 있을 때에는 점광원이 원의 형태로 투영됨을 볼 수 있다.

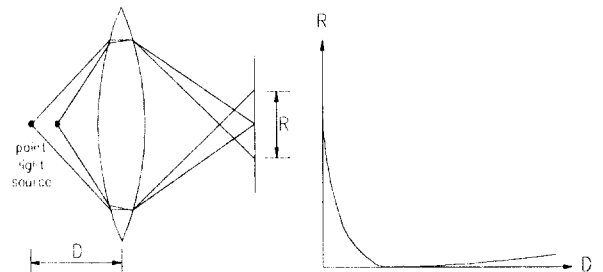


그림 3. 확산원의 정의와 지름의 변화
Fig. 3. Definition of Circle of Confusion and variation of its diameter

확산원의 모델은 여러 가지로 제시되어 왔다. 그 중 대표적으로 사용하는 것은 확산원 내부의 밝기 분

포가 일정하다고 가정하는 것과 가우시안 분포를 가진다고 가정하는 것이다⁽⁴⁾. 전자의 경우 점확산함수와 그것의 주파수 응답을 식 (9), (10) 에 나타내었으며 그림 4 에 2 차원 공간 주파수 영역에서의 특성을 나타내었다.

$$h(x, y) = \begin{cases} \frac{1}{\pi r^2} & x^2 + y^2 \leq r^2 \\ 0 & elsewhere \end{cases} \quad (9)$$

γ : 확산원의 반지름

$$H(m, n) = J_1\left(\frac{r\rho}{\rho}\right), \rho^2 = m^2 + n^2 \quad (10)$$

J_1 : 일차 베셀 함수

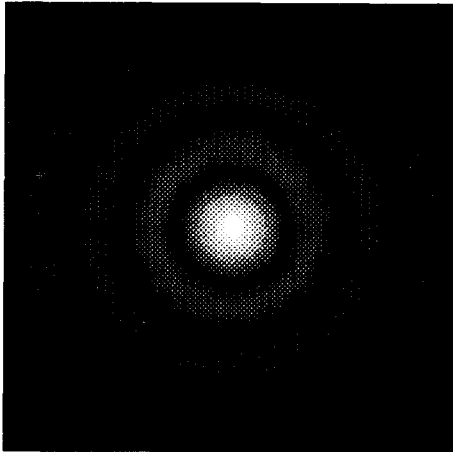


그림 4. 확산원 내부 밝기가 균일한 경우의 2 차원 주파수 영역 특성.
Fig. 4. Characteristic in 2 dimensional frequency domain in the case of constant brightness in circle of confusion.

위의 식에 의하면 확산원의 경계에서 불연속성이 생기는데 실제로는 회절이나 렌즈의 수차 등으로 인하여 그 경계가 명확하지 않으며 내부 밝기도 균일하지 않다. 따라서 종종 다음과 같은 2차원 가우시안 함수로 모델링을 하기도 한다⁽⁵⁾.

$$h(x, y) = \frac{1}{2\pi \sigma_h^2} e^{-\frac{x^2 + y^2}{2\sigma_h^2}} \quad (11)$$

$$H(m, n) = e^{-\frac{m^2 + n^2}{2\sigma_h^2}} \quad (12)$$

식 (11)은 점확산함수이며 그것의 주파수 응답이 식(12) 이고 그림 5는 2차원 공간 주파수 영역에서의 특성을 도시한 것이다. σ_h 는 확산계수라고 부르며 확

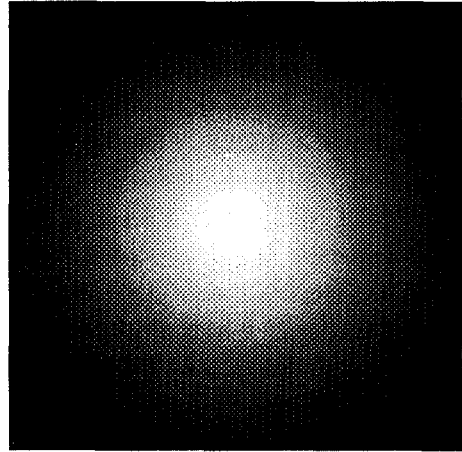


그림 5. 확산원 내부 밝기가 가우시안 분포인 경우의 2차원 주파수 영역 특성
Fig. 5. Characteristic in 2dimensional frequency domain in the case of Gaussian brightness in circle of confusion

산원에 비례하는 값인데 비례상수는 촬상시스템에 따라 달라진다.

식 (10)과 (12)에서 볼 수 있듯이 초점이 맞지 않아서 생기는 흐림은 저역 통과 여파기의 특성을 지닐 수 있다. 또한 확산원의 크기가 커짐에 따라 통과하는 대역이 점점 낮아짐을 알 수 있다. 따라서 초점이 맞지 않아 생기는 흐림현상의 정도는 고주파 성분의 크기로서 알아낼 수 있다.

움직임 흐림 현상은 카메라의 셔터 속도와 촬상 소자의 감도가 유한하다는 데에 기인한다. 3차원 피사체가 촬상기 내부의 2차원 촬상면에 투영될 때 3차원 피사체 표면의 각 점이 2차원 촬상면의 각 점으로 대응된다. 촬상면 위의 각 점은 자신에게 대응된 점의 밝기를 셔터가 열린 시간동안 누적하여 출력한다. 셔터가 열려 있는 동안 피사체가 움직이면 피사체 표면의 각 점은 촬상면위의 한 점에 대응되지 않고 대응되는 점들은 궤적을 이루게 되어 화상은 흐르는 것 같이 보인다. 피사체의 움직임보다 셔터의 속도가 충분히 빨라 셔터가 열린 동안 피사체 표면의 각 점이 촬상면에 남긴 궤적을 직선으로 간주할 수 있다고 가정하면 초점이 맞지 않아 일어나는 흐림 현상에서는 피사체 표면의 한 점이 촬상면에 원으로 대응되지만 이 경우에는 선분으로 대응된다. 피사체가 일정한 속도 V 로 움직이고 있고 카메라의 노출이 T 시간 동안

이러면 촬상된 화상은 움직임 방향으로 흐림 현상이 발생하고 점확산함수는 다음과 같이 주어지게 된다.

$$h(x, y, s, t) = h(x-s) = \begin{cases} \frac{1}{VT} \delta(y-t) & 0 \leq x-s \leq VT \\ 0 & \text{otherwise.} \end{cases} \quad (13)$$

이산 신호의 경우에는, 점광원이 L길이의 선분에 대응되는 흐림 현상이 나타날 때, 점확산함수및 그것의 주파수 응답을 다음과 같이 나타낼 수 있다.

$$h(i, j, k, l) = h(i-k) = \begin{cases} \frac{1}{L+1} & 0 \leq i-k \leq L, j=l \\ 0 & \text{otherwise.} \end{cases} \quad (14)$$

$$H(m, n) = \frac{1}{L+1} e^{-j(L+1)nm} \frac{\sin(\pi(L+1)m/N)}{\sin(\pi m/N)} \quad \forall n \quad (15)$$

그러므로, 운동 방향을 x 축이라 할 때 움직임 흐림은 x축 방향의 고주파 성분을 감쇄시키는 (운동의 수직 방향에 대해서는 평탄한 특성을 가지는) 여파 효과를 가진다. 이 여파 효과를 x축을 따라 1차원으로 생각하여 2차원 공간 주파수 영역에서의 특성을

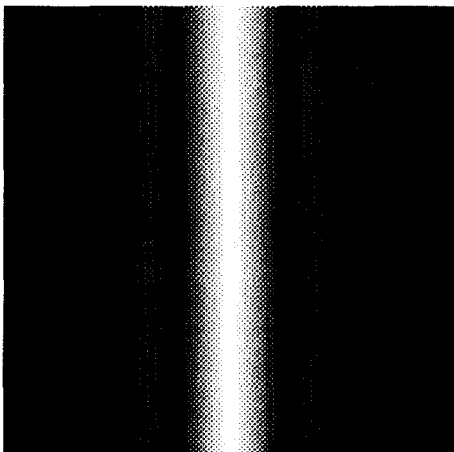


그림 6. 움직임 흐림의 2 차원 주파수 영역 특성
Fig. 6. Characteristic in 2dimensional Frequency domain in the case of motion blur

나타낸 것이 그림 6이다. 따라서, 움직임 흐림이 초점이 맞지 않은 경우의 흐림과 구별될 수 있는 점은 바로 방향성이 있다는 것이다.

초점이 안 맞아 흐린 경우는 화상의 질을 떨어뜨리지만 움직임 흐림의 경우는 인간이 동화상을 보는 데에는 오히려 유리한 면이 있다. 촬상면위의 어느 한 점은 시간에 대해 1차원 신호인 빛의 플럭스(flux)를 입력으로 받는다. 만약, 셔터 속도가 무한히 빠르고 촬상 소자의 감도가 무한하다면 촬상 소자의 출력은 입력을 이상적으로 표본 추출한 결과가 되고 이때 시간에 대한 aliasing이 생긴다. 그러나, 셔터 속도가 유한하면 입력을 일정 간격 동안 평균한 신호를 표본 추출한 결과가 된다. 이때 평균은 저역 통과 여파기의 역할을 하므로 antialiasing 효과를 갖게 되고 인간이 동화상을 보는 데에는 오히려 더 유리하게 된다. 그러나, 자동초점조절 장치에게는 고주파 성분이 줄어든 결과가 되어 초점이 안 맞은 것으로 판단할 수 있다. 예를 들어 농구 경기와 같은 빠른 움직임을 촬영한다고 할 때 움직임 흐림으로 인하여 초점값이 계속 변동을 하지만 카메라와 피사체와의 거리가 크게 변하지 않았다면 계속 합초점 상태를 유지하여 주는 것이 좋다. 또한 카메라를 빠르게 패닝시키는 경우에는 사람 눈이 공간 해상도에 대해 둔감해지므로 도중에 피사체의 거리가 변하더라도 그 초점상태를 유지하는 것이 합초점을 찾기 위해 렌즈가 움직이는 것보다 더 좋게 느껴진다. 그러므로, 시각적으로 편안하게 느껴지도록 초점을 맞추기 위해서는 움직임 흐림 현상에 대한 고려가 필요하다.

Ⅲ. 화상의 특징을 반영하는 초점값

흐림 상태가 심한 경우에는 고주파 성분이 거의 없다. 흐림 상태에서 초점이 맞아간다는 것은 세밀한 무늬보다는 윤곽이 선명해져감을 의미한다. 이 윤곽의 주파수 성분은 대역이 낮다. 그러나 합초점 부근에서 정밀하게 초점을 맞추는 경우에는 윤곽성분보다는 세밀한 무늬쪽이 영향을 많이 주게 된다. 그러므로 흐림 정도에 따라 초점값 계산에 사용하는 주파수 성분의 대역이 달라져야한다. 이상적으로는 다수의 여파기를 준비하여 초점 정도에 따라 전환하여 사용하는 것이 좋겠으나 하드웨어가 복잡해지고 연산시간

이 길어지게 되므로 본 논문에서는 보다 간단한 형태의 여파기를 사용하였다.

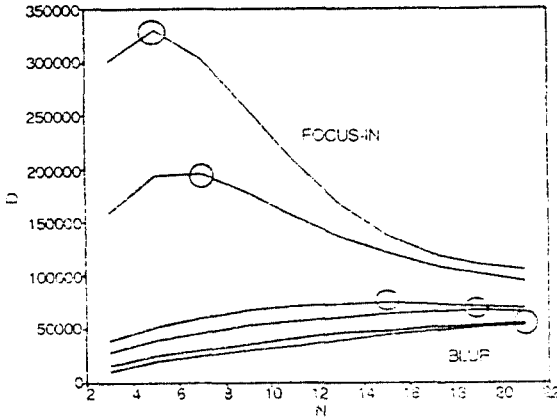


그림 7. 흐림 정도에 따른 주파수 대역별 초점값 변화율
Fig. 7. Changing rate of focus value with regard to blurring state

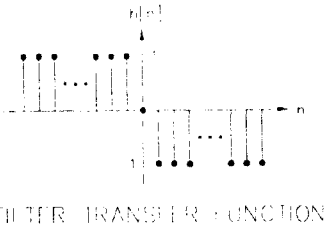


그림 8. 여파기 충격 응답
Fig. 8. Impulse response of the filter

그림 7은 흐림 정도에 따라 여러가지 주파수 대역을 이용하는 초점값의 변화율이다. 가로축은 여파기의 차수이고 세로축은 초점값의 크기이다. 여파기는 그림 8 형태의 것을 사용하였다. 이 형태의 여파기는 차수가 커질수록 통과 대역이 낮아지게된다. 그림 7에서는 각각의 흐림 정도에서 변화율이 가장 큰 차수에 원을 표시하였다. 흐릴수록 낮은 주파수 대역을 초점값 계산에 이용하는 것이 효과적임을 알 수 있다. 실제 칩 구현에는 그림 8의 형태를 직렬 2단으로 나누어 평균을 구하는 전단과 차이를 구하는 후단으로 구현하였다. 평균을 구하는 전단에서 화소들의 평균을 구한 다음 메모리에 저장하고 그 결과를 후단에서 임의의 간격으로 빼는 것이다. 빼는 간격이 크다는 것은 곧 여파기의 차수가 증가함을 의미한다. 이러한 여파기의 특성 및 전달함수에 대해서는 참고문헌

[6-8], [11,12]에 자세히 나와 있으므로 본 논문에서는 생략하기로 한다.

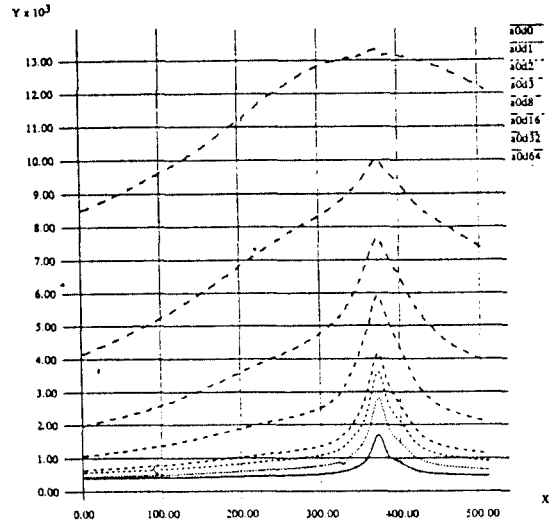


그림 9. 초점값 곡선. 평균하는 화소수=1
Fig. 9. Focus Value Curve. Average pixel=1

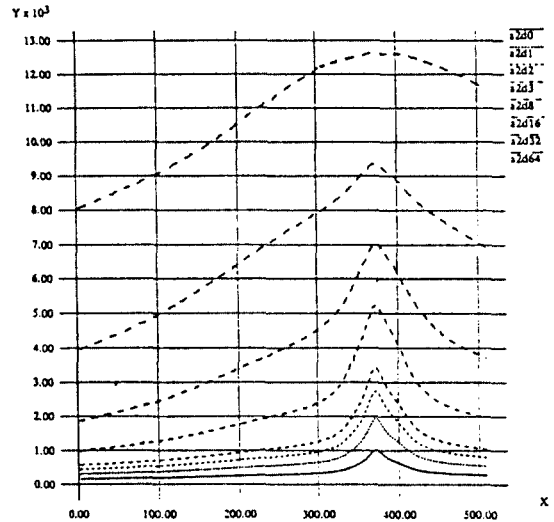


그림 10. 초점값 곡선. 평균하는 화소수=4
Fig. 10. Focus Value Curve. Average pixel=4

그림 9는 초점값의 그래프이다. 가로축은 초점이 맞아가는 단계를 나타내고 그래프 우측의 각 선들의 이름은 초점값 방식을 나타낸다. 'a' 뒤의 숫자는 평균을 취하는 화소의 수, 'd' 뒤의 숫자는 차이를 구할

때 빼는 간격의 수이다. 그래프를 보면 빼는 간격이 작을 때에는 흐림 상태에서는 초점값 변화가 없다가 초점이 거의 다 맞아가면서 급격한 변화가 생긴다. 그러나 빼는 간격이 클 때에는 흐림 상태에서도 초점값 변화가 있고 초점이 어느 정도이상 맞으면 초점값 증가가 둔화된다.

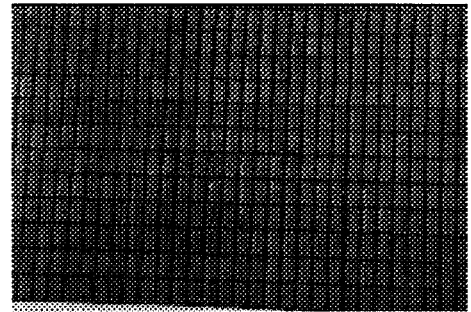
화소들끼리 평균을 취함으로써 가산성 잡음에 강하게 할 수 있다. 그림 9와 그림 10을 비교해 보면 그림 10의 경우가 곡선의 모양이 깨끗함을 알 수가 있다. 그림 9에서는 초점값이 증가하는 도중에 국부적인 최대값을 보인다. 이런 경우에 이 지점을 합초점 위치로 잘못 판단할 우려가 있으므로 화상잡음이 심한 경우 평균을 취하는 거리를 크게 하는 것이 유리하다.

이와 같은 구성의 여파기는 하드웨어적으로 간단하고 도중에 평균하는 화소수및 빼는 간격을 변화시킴으로서 특성을 바꾸어 줄 수 있다는 장점이 있다. 흐림이 심한 경우에는 평균을 취하는 화소수와 빼는 간격을 크게 하고 점점 초점이 맞아갈 수록 그 값들을 줄여감으로서 좋은 결과를 얻을 수 있었다.

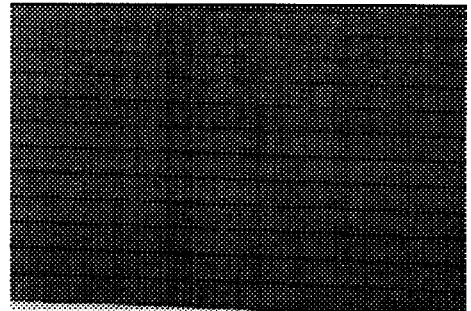
IV. 움직임의 추정

움직임 흐림의 경우 운동 방향으로만 흐려진다는 점을 이용하여 움직임을 추정할 수 있다. 운동방향과 수직이 되는 방향은 움직임 흐림에 의한 영향을 받지 않으므로 콘트라스트가 어느 방향으로 존재하는지를 알아보면 운동의 방향을 알 수 있다⁹⁾. 그림 11은 움직임 흐림으로 생긴 방향성 콘트라스트의 예이다. 그림 11의 (a)는 정지해 있는 상태를 촬영한 것이고 (b)는 카메라를 수평방향으로 패닝시키면서 촬영한 것으로서 (a)와는 달리 가로줄무늬가 발생함을 알 수 있다. 콘트라스트 방향 정보를 지니면서 누적했을 때 서로 다른 방향끼리는 상쇄되고 같은 방향은 보강되는 어떤 양을 만들면 한 영역에서 콘트라스트에 방향성이 있는지를 결정하는 기준으로 사용할 수 있다. 이를 위해 픽셀 (x,y)에서의 그래디언트 벡터 v(x,y)를 식 (16)와 같이 정의한다.

$$\begin{aligned} x &= f(x+1, y) - f(x-1, y), \\ y &= f(x, y+1) - f(x, y-1). \end{aligned} \quad (16)$$



(a) 정지상태의 벽, (a) Wall in still



(b) 패닝도중의 벽, (b) Wall in panning

그림 11. 움직임 흐림의 효과
Fig. 11. Effect of motion blur

방향성의 연관정도를 알기 위한 간단한 방법은 그 그래디언트 벡터를 평균한 벡터를 구하는 것이다. 벡터를 평균한다는 의미는 벡터의 원소각각을 평균한다는 것이다. 그러나 단순한 평균만으로는 실질적인 방향성의 연관정도를 알 수 없다. 예를 들어 서로 반대 방향을 가리키는 벡터를 생각해 보자. 이 두 벡터를 더하게 되면 0이 되지만 화상내에서 두 벡터는 같은 방향을 의미하게 된다(그림 12 참조). 즉 단순한 평균의 결과로는 화상내에서 일관된 방향성을 갖지 않는 것처럼 되나 실제로는 단일한 방향성을 갖는다. 이와는 반대로 서로 수직인 벡터는 전혀 상관성이 없음에도 더했을 때 0이 되지 않는다.

이 문제를 해결하기 위해서 그래디언트 벡터의 각도를 2배한 다음 평균을 취하는 방법을 택하였다. 이 경우에 서로 반대방향의 벡터와 수직인 벡터의 연산 결과를 그림 13과 식 (17)에 보였다.

반대방향의 경우

$$\begin{aligned} &(\gamma \cos 2(\theta + \pi), \gamma \sin 2(\theta + \pi)) + (\gamma \cos 2\theta, \gamma \sin 2\theta) \\ &= (\gamma \cos 2\theta, \gamma \sin 2\theta) + (\gamma \cos 2\theta, \gamma \sin 2\theta) \end{aligned}$$

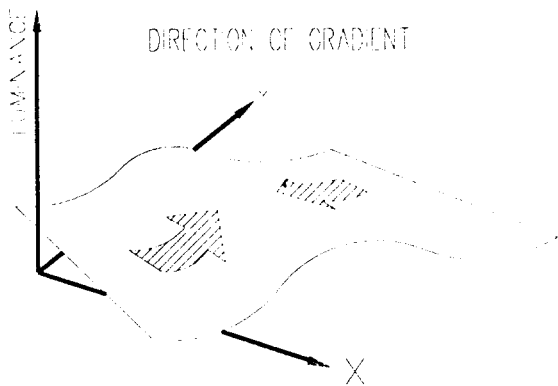


그림 12. 동일한 방향성을 나타내는 반대방향의 벡터
 Fig. 12. Two opposite vectors representing the same directional property of pattern

$$\begin{aligned}
 &= 2(\gamma \cos 2\theta, \gamma \sin 2\theta), \\
 \text{수직방향의 경우} & \quad (17) \\
 &(\gamma \cos 2(\theta + \frac{\pi}{2}), \gamma \sin 2(\theta + \frac{\pi}{2})) + (\gamma \cos 2\theta, \gamma \sin 2\theta) \\
 &= (-\gamma \cos 2\theta, -\gamma \sin 2\theta) + (\gamma \cos 2\theta, \gamma \sin 2\theta) \\
 &= (0, 0)
 \end{aligned}$$

이 때 2배각 그래디언트 벡터의 평균의 크기는 연관성의 정도를 나타낸다. 그 값이 클수록 화상내에서 동일한 방향을 나타내는 벡터가 많다는 의미가 된다. 그림 14는 움직이는 사람이 계속 화면의 가운데에 위치하도록 카메라 패닝을 하는 도중의 한 장면이다. 이 경우에 실제로는 사람이 움직이는 것이지만 화면 상에 나타나는 현상은 배경이 수평으로 흐르는 것처럼 나타난다. 그림 15와 그림 16는 화면을 분할하여 각 부분 블럭내에서 2배각 그래디언트 벡터의 평균을 나타낸 것이다. 원래의 그래디언트 벡터는 콘트라스트 패턴에 수직이 되지만 패턴의 방향과 일치하도록 하기 위해 90도 돌려서 나타내었다. 또한 평균을 구할 때 각을 2배 하였으므로 평균을 취한 다음에는 각을 2로 나누어주었다. 각 분할 영역의 크기가 작은 그림 15에서는 배경부분에서 그래디언트 벡터가 수평 방향으로 존재하여 수평 방향의 운동성분이 있음을 알려준다. 그러나 사람부분에 있어서는 사람의 윤곽 자체의 모양을 나타냄을 알 수 있다. 분할 영역의 크기가 큰 그림 16에서는 배경의 경우 그림 15와 동일하게 수평방향의 운동성분을 보여주지만 사람부분에서는 거의 0 벡터를 나타낸다. 이것은 사람의 윤곽

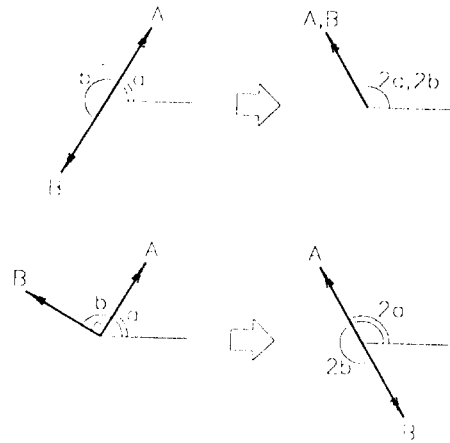


그림 13. 벡터의 각 2배 만들기
 Fig. 13. Doubling the angle of a vector

을 나타내는 벡터들이 서로 상관성이 없으므로 평균적으로 0 이 되기 때문이다. 따라서 분할 영역을 크게 할 경우에 카메라의 패닝과 같은 화면 전체의 이동을 파악하기 쉽고 분할 영역을 작게 하는 경우에는 국부적인 특성을 파악할 수 있으므로 화면 내 일부분의 움직임이나 피사체 존재 자체를 파악하는데 적용될 수 있다.



그림 14. 참고있는 피사체
 Fig. 14. Image of a walking woman

이러한 움직임 추정 방법은 기존에 알려진 다른 방법들에 비해서 정밀도는 떨어지지만 간단한 하드웨어로 빠른 시간안에 수행할 수 있으므로 실시간, 즉 1/60초 이내에 피사체나 화면 전체의 운동여부 및 움직임 방향, 움직임의 크기 등을 알 수 있다. 이러한 정보를 이용함으로써 화면에 흐림이 발생할 경우 움직임으로 인한 것인지 초점이 맞지 않아서 그런 것인지를 알 수 있으며 일단 초점이 맞은 상태에서 피사

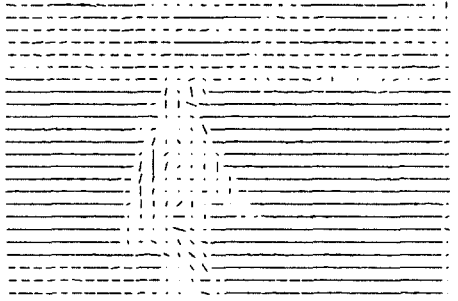


그림 15. 2 배각 그래디언트 벡터의 부분 블록의 합(분할영역의 크기가 작을 때)
Fig. 15. Local averages of angle-doubles gradient vectors (small subblock)

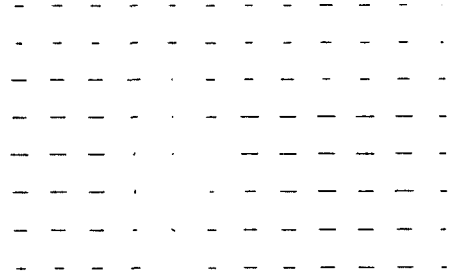


그림 16. 2 배각 그래디언트 벡터의 부분 블록의 합(분할영역의 크기가 클 때)
Fig. 16. Local averages of angle-doubles gradient vectors (large subblock)

체가 이동하거나 패닝의 경우에도 대처할 수 있다.

움직임 추정 방법을 하드웨어로 구현하는데 있어서 문제가 되는 것은 그래디언트 벡터의 각을 2배로 만드는 부분이다. 이를 위해서 그래디언트 벡터를 복소수로 보고 그 값을 제공하면 각은 2배가 되고 크기는 제공이 된다. 그림 17은 이 과정을 도해한 것이다. 이 때 크기를 다시 원래의 크기로 만들기 위해서 식 (18)과 같이 원래 벡터의 크기로 나누어주어야 한다.

$$x' = \frac{x^2 - y^2}{\sqrt{x^2 + y^2}}, y' = \frac{2xy}{\sqrt{x^2 + y^2}} \quad (18)$$

$$v'' = v^2 = (x^2 - y^2) + i(2xy)$$

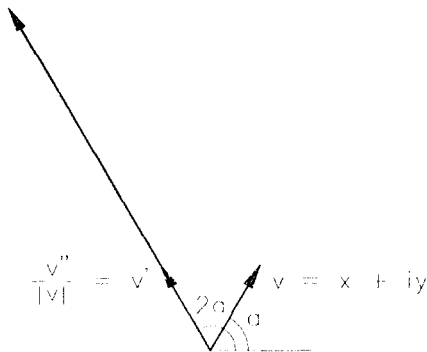


그림 17. 각을 2배하기 위한 복소수의 제곱
Fig. 17. Squaring of a complex number to double the angle

그런데 제공근을 구하는 부분과 나누는 부분은 하드웨어의 면적이나 계산 시간상으로 많은 부담을 주게 되므로 실제 하드웨어 구현에서는 로그(logarithm)와 같은 비선형 함수를 이용하여 다이내믹 레인지를 줄이

는 방법을 사용하였다. 따라서 실제로 구하는 것은 다음 식과 같으며 이를 '방향 벡터'라고 부른다. 식 (19)를 구현하기 위해서는 2개의 제공기와 1개의 곱셈기가 필요하며 그림 18에 그 구성을 나타내었다.

$$x' = x^2 - y^2, y' = 2xy \quad (19)$$

방향 벡터를 평균함으로써 움직임이 있는지의 여부는 쉽게 판단할 수 있다. 그러나 그 움직임이 화면 전체에서 일어난 것인지 아니면 국부적인지를 판단하기 위해서는 분할 영역 모두에 대한 정보를 가지고 있어야 한다. 제작한 칩에서는 화면분할의 정보를 내부적으로 저장하고 외부에서 요구가 있을 때 그 정보를 전달해 줄 수 있도록 설계되었지만 실제 비디오 카메라에서는 제어용 마이크로 프로세서와 칩간의 통신가능 시간이 매우 짧고 또 통신대역폭이 넓지 못하므로 분할 영역전체의 정보를 전달해 줄 수 없는 경우가 있다. 이러한 경우에서도 움직임을 판단하기 위해 분할 블록에서 구해진 평균 방향벡터들의 평균과 분산을 이용하는 방법도 개발하였다. 예를 들어 화면 전체의 평균이 동일하다라도 패닝과 같은 전반적인 움직임이라면 분산은 작게 되고 국부적인 움직임이라면 분산이 크게 된다.

움직임의 추정은 화면의 상태를 모르고서는 정확히 추정하기 어렵다. 또한 평균과 분산도 절대적인 값이 중요한 것이 아니라 현재 화면 상태에 따른 상대적인 값으로 고려해야 한다. 따라서 실제 마이크로 프로세서에서 판단을 내릴 때는 영역별 축점 평가치 및 밝기를 방향 벡터의 평균, 분산과 함께 고려한다.

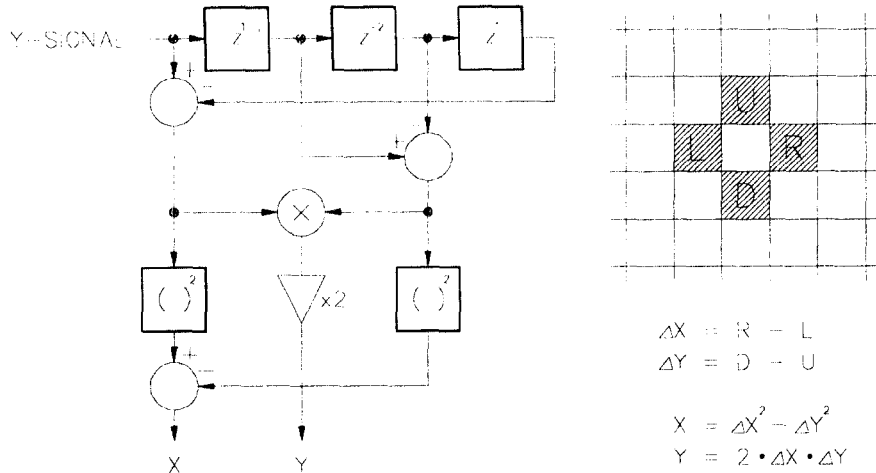


그림 18. 각을 2배하는 부분의 블럭도
Fig. 18. Block diagram of the angle-doubling block and the gradient calculation

V. 하드웨어 구현

본 논문에서 사용한 칩은 크게 세 가지의 기능을 하도록 설계되었다. 첫번째 기능은 자동초점조정 알고리즘에서 사용되는 초점값 및 화면의 상태를 알려주는 정보들을 구하는 것이다. 초점값은 기본적으로 인접 화소간의 휘도값의 차를 구하며, 수평 모드와 수직 모드로 나뉘어 진다. 수평 모드에서는 수평으로 인접한 화소간의 휘도값의 차를 구하거나 임의의 간격을 둔 화소간의 휘도값의 차를 구한다. 수직 모드에서는 현재의 화소값과 직전 주사선에서 수직으로 인접한 휘도값의 차를 구하여 누적한다. 두 모드에서 공통적으로 이동 평균(moving average)의 개념을 도입하여 시스템의 잡음이 심하거나 외부의 요구가 있을 경우에는 연속된 화소의 휘도값들의 평균값으로 신호 처리를 한다. 시스템 잡음이 심할 경우, 일정 값 이상은 무시할 수 있도록 문턱 값(threshold value)을 지정해 줄 수 있으며, 입력 휘도신호를 로그 변환을 하여 처리를 할 수 있다. 초점값을 화면의 영역별로 구하면, 움직임 판별 등의 유용한 정보를 얻을 수 있다. 본 시스템에서는 전체 화면을 수평으로 8등분하고 수직으로 8등분하여 총 64개의 영역에 대해서 각각 초점값을 구한다. 한 영역은 수평으로 42화소, 수직으로 26화소의 크기를 갖는다.

칩의 두번째 기능은 퍼지 추론이다⁽¹⁰⁾. 퍼지 연산부의 사양은 다음과 같다. 연산 시간은 퍼지규칙 하나당

대략 70us정도 걸린다. 멤버십함수의 입력공간 크기는 64이고 해상도는 6bit이다. 어드레스 공간은 17bit으로 2044개의 멤버십을 액세스 한다. 64개까지의 연산을 처리할 수 있고 한 개의 연산은 64개의 규칙까지 포함할 수 있으며 규칙 하나는 64개의 입력까지 받을 수 있다. 퍼지 규칙은 외부 롬에 저장한다. 외부 롬과의 연결에는 17bit 어드레스 버스와 6bit 데이터 버스, 그리고 롬 enable 신호를 사용한다.

칩의 세번째 기능은 전체 칩의 데이터 흐름을 조절하며, 외부 시스템과의 통신을 담당하고, 부가로 외부 모터를 제어하는 기능이다. 이 기능을 하는 부분은 명령어 해독 블럭, 모터 제어, 초점값 연산부 제어, 퍼지 연산 제어, ADC제어, DAC제어 블럭 등으로 이루어 진다. 명령어 해독 블럭은 칩에 입력되는 명령어가 여러 제어 블럭 중 어느 것을 구동할 지를 결정하고 그에 따라 그 블럭에 제어 신호를 보내고 그 블럭에서 명령 수행이 완료의 신호를 받아 외부에 명령 수행이 끝났음을 알린다. 모터 제어 블럭은 모터의 모드와 모터 제어 명령어의 파라미터에 따라 여러가지 경우에 대해 동작하며 스텝 모터를 2개까지 동작시킬 수 있다. ADC제어 블럭은 칩 외부의 ADC에 아날로그 초점값, 줌값, AGC값, 조리개 조절값을 선택적으로 보내고 그에 대한 디지털 값을 받아들인다. DAC제어 블럭은 조리개 제어를 위해 병렬 입력을 받아 순차적으로 이를 내보내는 기능을 갖는다. 칩의 내부 버스는 초점값 연산부, 퍼지 연산부와 함께 공유하는데 이 내

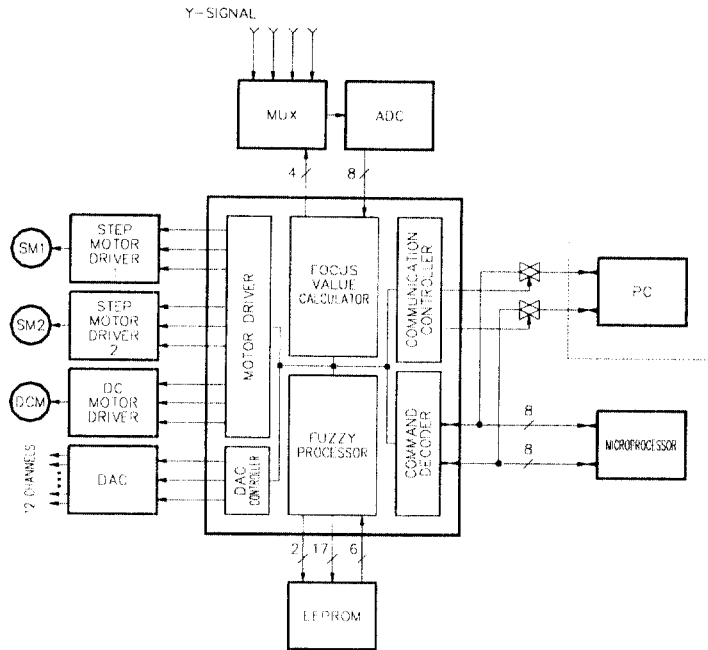


그림19. 칩의 개념도
Fig. 19. Block Diagram of the Chip

부 버스의 제어와 외부 마이크로 프로세서와 충전값 연산부, 퍼지 연산부의 연결을 이들의 제어 블럭이 맡는다.

칩은 COMPASS 를 이용하여 설계하였으며 0.8um CMOS 게이트 어레이로 제작되었다. 제작된 칩의 크기는 $5 \times 5\text{mm}^2$ 이고 100개의 핀을 갖는다. 그림 19는 설계한 칩의 개념도이며 그림 20에 칩의 사진을 보였다.

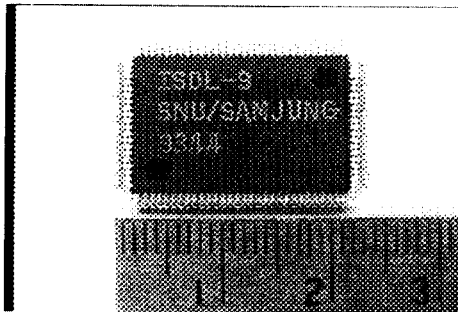


그림 20. 제작한 칩의 외형
Fig. 20. Photography of the Chip

VI. 실험결과 및 검토

제작한 칩의 성능을 검증하기 위하여 시중에 판매 중인 캠코더를 구입하여 그 중에서 자동충점조절 기능을 수행하는 칩과 마이크로 프로세서를 제거하고 제작한 칩과 마이크로 프로세서로 구성된 제어보드를 부착하여 다양한 상황에서 촬영을 해 보았다. 그림 21, 그림 22는 제어보드의 사진 및 실험장치의 구성도이다. 제어보드의 마이크로 프로세서와 PC는 제작한 전용 병렬 인터페이스 보드를 이용하여 통신할 수 있

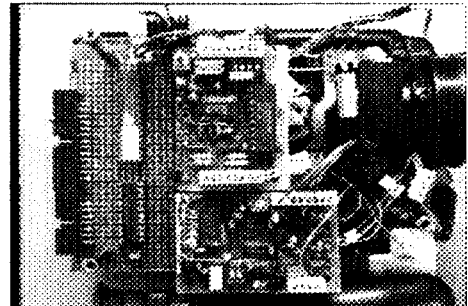


그림 21. 제어보드의 사진
Fig. 21. Photography of a control board

도록 하였다. 이를 이용하여 PC의 모니터 화면상에 초점값 곡선이나 64 분할된 영역별 정보를 볼 수 있도록 하였으며 반대로 알고리즘의 여러 파라미터들을 PC에서 제어보드의 마이크로 프로세서로 전송할 수 있도록 하였다. 즉 PC는 현재의 상황을 알아보거나 마이크로 프로세서에 명령을 내릴 때 사용할 뿐 실제의 모든 연산 및 제어는 캠코더에 부착된 제어보드의 칩과 마이크로 프로세서에서 수행된다.

실험결과 기존의 방식으로는 초점을 맞추기 어렵거나 정밀한 초점을 유지하지 못했던 많은 경우에 좋은 결과를 얻을 수 있었다. 그림 23, 24 는 수직 창살뒤의 사람을 촬영한 것이다. 이 경우 기존의 방식에서는 수평방향의 초점값만을 구하므로 수평 방향으로 큰 콘트라스트를 나타내는 창살에 초점이 맞게 된다. 그러나 본 시스템에서는 칩내부에 라인 메모리를 내장하고 수직 방향의 초점값도 구할 수 있으므로 이 상황을 판단하여 창살뒤의 피사체를 맞춘 것이다. 그림 25, 26은 피사체의 앞부분에 일정부분 방해물이 있는 경우이다. 이 경우도 기존의 방법으로는 앞의 커튼을 맞추게 되지만 화면의 영역별 분할 정보를 이용함으로써 사람들이 일반적으로 맞추고 싶어하는 뒷부분을 맞출 수 있다. 그림 27, 28은 깜박이는 크리스마스트리를 촬영한 것인데 기존의 방법에서는 광원이 초

점을 맞추는데 방해를 하고 또한 깜박이는 상황이 초점값에 커다란 교란작용을 하므로 초점을 맞출 수 없는데 반해 본 시스템은 영역별 정보와 여러가지 초점값의 조합을 이용하여 초점을 맞출 수 있었다.

움직임 정보를 이용한 예는 그림 29에 나타나 있다. 앞의 피사체는 정지해 있지만 뒤의 자동차들은 빠르게 수평방향으로 움직이고 있다. 이 경우에 각 부분 블럭의 움직임 정보를 알아냄으로서 앞의 피사체는 정지해 있음을 알 수 있고 따라서 초점을 맞추는 영역을 가운데 부분으로 축소시켜 앞의 피사체에 초점을 맞추고 계속 합초점 상태를 유지할 수 있었다. 그림 30은 패닝도중의 장면이다. 이 경우 흐림 현상이 상당히 나타나 있음을 알 수 있다. 초점값만을 추적하는 경우라면 이 상태에서 화면이 변화한 것으로 판단하고 합초점위치를 찾으려고 렌즈를 구동시키게 되나 본 시스템에서는 패닝 도중인 것으로 판단하여 렌즈의 위치를 고정시키고 패닝을 멈춘 후에는 피사체에 대해 합초점위치를 찾았다. 패닝 속도가 느린 경우에는 패닝 도중에도 초점조절을 해주어야 하는데 이 경우에도 화면 전체의 움직임 정도를 알아냄으로서 대응할 수 있었다.

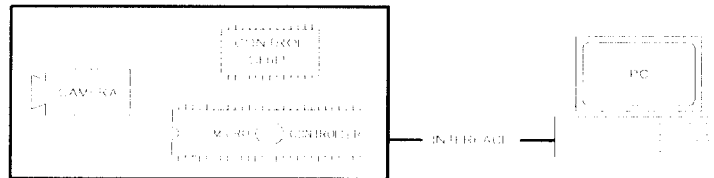


그림 22. 실험장치의 구성도
Fig. 22. Configuration experimental device



그림 23. 수직 창살뒤의 사람(기존 방법)
Fig. 23. Man Behind vertical Bar(Conventional System)



그림 24. 수직 창살뒤의 사람(제안 방법)
Fig. 24. Man Behind vertical Bar(Proposed System)



그림 25. 커튼에 가린 인형(기존 방법)
Fig. 25. Doll with Interfering Curtain(Conventional System)



그림 26. 커튼에 가린 인형(제안 방법)
Fig. 26. Doll with Interfering Curtain(Proposed System)

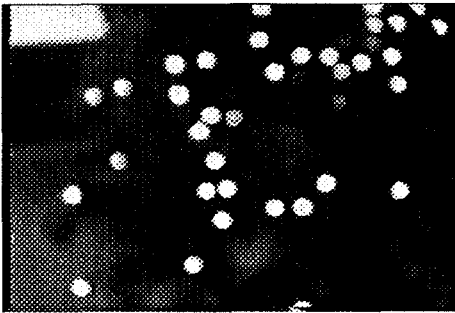


그림 27. 깜박이는 전구(기존 방법)
Fig. 27. Flickering Bulbs(Conventional System)

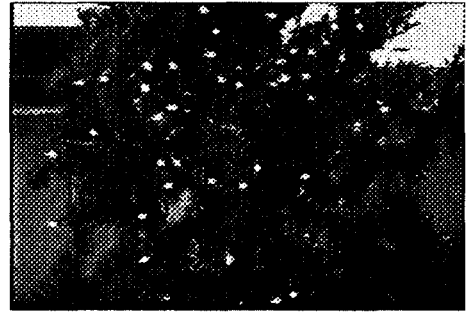


그림 28. 깜박이는 전구(제안 방법)
Fig. 28. Flickering Bulbs(Proposed System)



그림 29. 달리는 자동차앞의 사람
Fig. 29. Man front of running cars



그림 30. 패닝도중의 피사체
Fig. 30. Objects in panning

Ⅶ. 결 론

추적값만을 추적하는 기존 자동추적조절 방법의 한계를 극복하기 위하여 화면의 영역분할을 통해 화면 상태를 파악하고 움직임 상태를 알 수 있는 알고리즘을 개발하였다. 또, 상황에 따라 화상의 주파수 성분이 크게 달라지므로 고정된 주파수 대역만을 이용하는 것이 아니라 대역을 전환하면서 다양한 종류의 추적값을 적용할 수 있도록 하였으며 다양한 상황 판단

을 최소한의 하드웨어 및 소프트웨어 부담으로 처리하기 위해 퍼지 보조 연산기를 칩 내부에 내장하여 추적값 연산과는 독립적으로 수행할 수 있도록 하였다.

위와 같은 기능을 가진 칩을 제작하여 자동추적조절 장치를 구성하고 다양한 상황에서 실제로 촬영을 함으로서 성능을 검증하였다. 화면 분할과 움직임 정보를 이용하여 퍼지 추론을 함으로서 기존의 자동추적조절 방법으로는 추적을 맞추기 곤란했던 피사체에

대해서 초점을 맞출 수 있음을 확인하였다.

현재 화면 분할을 좀 더 세분화하고 다른 블럭들의 기능도 향상시킨 칩을 제작중에 있다. 화면 분할을 더 세분화함으로써 제작한 칩을 자동초점조절이외에 피사체 인식, 피사체 추적등에 사용할 수 있을 것으로 생각된다.

참고문헌

1. 이 충웅 외, "Camcorder 용 Auto Focusing System 의 전용 IC 및 경통부 개발에 관한 연구." 상공부 공업기반기술개발사업 기술개발 최종 보고서, 상공부, 1993.
2. Eric Krotkov, "Focusing," International Journal of Computer Vision, 1, pp.223-237, 1987.
3. Jan Biemond, Reginald L.Lagendijk and Russell M. Mersereau, "Iterative Methods for Image Deblurring," Proceedings of IEEE, Vol. 78, No. 5, pp. 856-883, May 1990.
4. M. Subbarao, T. Choi and A. Nikzad, "Focusing Techniques," Optical Engineering, Vol.32, No. 11, pp. 2824-2836, Nov. 1993.
5. Shree K. Nayar and Yasuo Nakagawa, "Shape from Focus," IEEE Transactions on PAMI, Vol. 16, No. 8, pp. 824-831, Aug. 1994.
6. 문 성진, "자동초점조절을 위한 흐림현상의 종류

판별 및 파라미터 추출에 관한 연구," 서울대학교 전자공학과 석사 졸업 논문, 1993.

7. 정 기욱, "화면 적응형 자동초점조절 프로세서의 설계에 관한 연구," 서울대학교 전자공학과 석사 졸업 논문, 1994.
8. Seong I. Jang, Jae. K. Chong, Sung H. Lee, Min H. Song, Ook Kim and Dong S. Kim, "Autofocusing Device for Use in a Video Camera and an Autofocusing Method thereof," U.S. Patent 5,200,828.
9. Seong-Ik Jang, Jaekyung Chong, Yongman Lee, Keewook Chung, Wonchan Kim, and Choong-Woong Lee, "A Real-Time Identification Method on Motion and Out-of-Focus Blur for a Video Camera," IEEE Trans. on Consumer Electronics, Vol. 40, No. 2, pp. 145-153, May 1994.
10. Yongman Lee, Seong-Ik Jang, Keewook Chung, Dongyun Lee, Wonchan Kim, and Choong-Woong Lee, "A Fuzzy-Control Processor for Automatic Focusing," IEEE Trans. on Consumer Electronics, Vol. 40, No. 2, pp. 138-144, May 1994.
11. 장 성익 외, "비디오 카메라의 초점값 생성방법", 특허 출원 제 11069 호, 1994.
12. 장 성익 외, "비디오 카메라의 초점값 생성방법 및 장치", 특허 출원 제 12938 호, 1994.

張成翼 (Seong Ik Jang)

1967년 4월 25일생.

1990년 2월 : 서울대학교 공과대학 전자공학과 졸업(공학사)
1992년 2월 : 서울대학교 대학원 전자공학과 졸업(공학석사)
1992년 3월~현재 : 서울대학교 대학원 전자공학과 박사과정 재학중

* 주관심분야 : 영상신호처리, 영상코딩, 비선형 필터링 등

李龍萬 (Yong man Lee)

1958년 4월 30일생

1981년 : 서울대학교 공과대학 전자공학과 졸업(공학사)
1984년 : 서울대학교 대학원 전자공학과 졸업(공학석사)
1995년 : 서울대학교 대학원 전자공학과 졸업(공학박사)
1984년~현재 : LG 반도체 근무중

* 주관심분야 : 고속회로 설계, 화상신호처리칩 등

鄭 起 旭 (Ki wook Chung)

1969년 7월 15일생

1992년 2월 : 서울대학교 공과대학 전자공학과 졸업(공학사)
1994년 2월 : 서울대학교 대학원 전자공학과 졸업(공학석사)
1994년 3월~현재 : 서울대학교 대학원 전자공학과 박사과정 재
학중
* 주관심분야 : 디지털신호처리, 화상신호처리칩 등

金 元 燦 (Won chan Kim)

1945년 11월 7일생

1972년 : 서울대학교 전자공학과 졸업(공학사)
1976년 : 독일 아헨 대학교 전기공학과 석사학위 취득
1981년 : 독일 아헨 대학교 전기공학과 박사학위 취득
1982년~현재 : 서울대학교 전자공학과 교수
* 주관심분야 : 기억소자, 데이터 변환회로, 퍼지칩, 카오스
칩, 화상신호처리칩 등

李 忠 雄 (Choong Woong Lee)

1935년 5월 3일생

1958년 3월 : 서울대학교 공과대학 전자공학과 졸업(공학사)
1960년 9월 : 서울대학교 대학원 전자공학과 졸업(공학석사)
1972년 7월 : 일본 동경대학교 공학박사 학위 취득
1964년 3월~현재 : 서울대학교 전자공학과 교수
1989년 1월~1989년 12월 : 대한전자공학회 회장 역임
1989년 1월~현재 : IEEE Fellow
1991년 6월~현재 : 서울대학교 부설 뉴미디어 공동연구소 소장
* 주관심분야 : 통신방식, 화상공학, TV방송시스템, 초고주
파공학, 의용공학, 미래통신 등