

# 비전 카메라와 다중 객체 추적 방법을 이용한 실시간 수질 감시 시스템

정희원 양원근\*, 이정호\*, 조익환\*, 진주경\*, 종신회원 정동석\*\*

## Real-time Water Quality Monitoring System Using Vision Camera and Multiple Objects Tracking Method

Won-Keun Yang\*, Jung-Ho Lee\*, Ik-Hwan Cho\*, Ju-Kyong Jin\* *Regular Members*  
Dong-Seok Jeong\*\* *Lifelong Member*

### 요약

본 논문에서는 비전 카메라와 다중 객체 추적 방법을 이용한 실시간 수질 감시 시스템을 제안하였다. 제안된 시스템은 기존의 센서 방식의 감시 시스템과 달리 비전 카메라를 이용해 객체를 개별적으로 분석한다. 비전 카메라를 이용한 시스템은 영상에서 개별 객체를 분리해 내는 방법과, 연속하는 두 프레임간의 상관관계에 의해서 다수의 객체를 추적하는 방법으로 구성된다. 실시간 처리를 위해 비모수 예측을 사용하여 배경 영상을 생성하고 이를 이용해 객체를 추출한다. 비모수 예측을 이용하면 연산량을 줄이는 동시에 비교적 정확하게 객체를 추출할 수 있다. 다중 객체 추적 방법은 개별 객체가 움직이는 방향, 속도 및 가속도를 이용해 다음 움직임을 예측하고 이를 기반으로 추적을 수행하였다. 또한 추적 성공률을 향상시키기 위해 예외처리 알고리즘을 적용하였다. 다양한 환경에서 실험한 결과 제안한 시스템은 처리 시간이 짧고 정확하게 다중 객체를 추적할 수 있어 실시간 수질 감시 시스템에 사용이 가능함을 확인하였다.

**Key Words** : Tracking, Multiple objects, Real-time, Non-parametric

### ABSTRACT

In this paper, we propose water quality monitoring system using vision camera and multiple objects tracking method. The proposed system analyzes object individually using vision camera unlike monitoring system using sensor method. The system using vision camera consists of individual object segmentation part and objects tracking part based on interrelation between successive frames. For real-time processing, we make background image using non-parametric estimation and extract objects using background image. If we use non-parametric estimation, objects extraction method can reduce large amount of computation complexity, as well as extract objects more effectively. Multiple objects tracking method predicts next motion using moving direction, velocity and acceleration of individual object then carries out tracking based on the predicted motion. And we apply exception handling algorithms to improve tracking performance. From experiment results under various conditions, it shows that the proposed system can be available for real-time water quality monitoring system since it has very short processing time and correct multiple objects tracking.

\* 인하대학교 전자공학과 멀티미디어 연구실 ({aida, julian, teddydino, jukyung77}@inhaian.net)

\*\* 인하대학교 전자공학과 교수 (dsjeong@inha.ac.kr)

논문번호 : KICS2006-11-517, 접수일자 : 2006년 11월 30일, 최종논문접수일자 : 2007년 4월 6일

## I. 서론

환경오염에 의해 인간이 받는 피해는 매우 심각하다. 때문에 최근 환경 호르몬과 관련된 문제들이 사회적으로 큰 이슈가 되고 있다. 다양한 환경오염 문제 중에서 수질 오염 문제는 인간의 생명과 직결된다는 점에서 매우 중요하다. 과거의 수질 감시 방법은 약품을 이용한 성분 분석 방법이 대부분이었다. 이 방법은 체계적이기는 하지만 오염 물질이 실제 생물체에 어떤 영향을 미치는지는 알 수 없다는 단점이 있다. 이런 이유로 최근에는 약품을 이용한 성분 분석 방법의 단점을 보완하기 위해 생물을 이용한 수질 감시 방법이 새롭게 제시되고 있다<sup>[1]</sup>. 그 중 물벼룩은 크기 및 운동성 측면에서 관찰하기 쉽다는 장점을 가지고 있다<sup>[2]</sup>. 생물에게 치명적인 오염원이 물벼룩이 담겨 있는 수조에 들어오면, 물벼룩은 특정한 반응(평균 이상의 빠른 움직임 또는 급격한 객체 수 감소 등)을 보이게 된다<sup>[3]</sup>. 이 반응을 실시간 감시하여 현재의 수질 확인 및 검사에서 누락된 오염원이 존재하는지를 확인할 수 있다. 물벼룩을 이용한 수질 감시 장치의 경우 하나의 수조 안에 여러 마리의 물벼룩이 움직이고 있기 때문에 각각의 객체에 대한 움직임을 개별적으로 추적하고 분석하는 과정이 필요하다. 기존 독일의 수질 검사 장비는 격자 모양의 센서를 이용하여 모눈종이 방식의 검출 알고리즘을 사용하였다<sup>[4]</sup>. 그러나 이 방법은 물벼룩의 전체적인 분포와 움직임 정도만 파악할 수 있을 뿐, 각각의 객체에 대한 분석을 수행할 수 없다. 하지만 카메라를 이용하면 입력된 영상을 분석한 후 개별 객체를 추적할 수 있다는 장점이 있다.

각각의 물벼룩을 개별 추적하는 방법을 이용하기 위해서는 우선 입력된 영상에서 물벼룩 객체를 추출해야 한다. 객체를 추출하기 위해서 비모수 예측을 이용해 배경 영상을 생성하고, 현재 입력 영상과 배경 영상과의 차영상을 구성한다. 비모수 예측을 이용한 방법은 확률에 기반을 둔 방법으로 조명의 변화에 의한 미세한 왜곡들을 최소화 할 수 있다는 장점이 있다. 추출된 객체를 추적하는 방법은 이전 프레임과의 상관관계에 의해 움직임을 예측하는 방법을 사용했다. 움직이는 물체는 같은 방향으로 계속해서 움직이려고 하는 힘, 즉, 관성이 존재하기 때문에 이전의 움직임을 분석하면 다음 움직임을 예측할 수 있다. 시스템을 구현하여 실험한 결과 객체 추출과 추적에서 비교적 높은 성공률을 보이는

것을 확인할 수 있었다. 또한 알고리즘의 수행시간이 비교적 짧아서 실시간 처리가 가능함을 확인할 수 있었다.

본 논문은 다음과 같이 구성되어 있다. 2장에서는 제안된 수질 감시 시스템의 구성에 대해 설명하고, 3장과 4장에서는 제안된 객체 추출 알고리즘 및 추적 알고리즘을 설명한다. 5장에서는 실험 결과에 대해 설명하고, 6장에서는 결론 및 향후 연구 과제에 대해 이야기 한다.

## II. 수질 감시 시스템의 구성

수질 감시 시스템은 실시간으로 상수원을 감시하는 장치이기 때문에 일반적으로 상수원 근처에 설치된다. 상수원으로부터 유입된 물은 감시 시스템을 통과해 다시 상수원으로 돌아가는 순환 과정을 거친다. 제안된 수질 감시 시스템은 물벼룩을 이용하기 때문에 유입된 물은 물벼룩이 담겨있는 수조를 통과하게 된다. 물벼룩의 행동을 분석하기 위해서 수조 위쪽에는 비전 카메라가 설치되어 실시간으로 물벼룩의 영상을 획득하고 이를 컴퓨터로 전송해 알고리즘을 수행한다. 물벼룩의 행동에 이상이 발생할 경우 컴퓨터와 연결된 경보 장치를 작동시켜 수질이 오염되었음을 작업자에게 알린다. 그림 1은 제안된 수질 감시 장치의 구성이다.

기존의 센서 방식의 감시 시스템은 물벼룩이 죽을 경우 바닥에 가라앉은 성질을 이용해 물벼룩의 유영 고도를 측정하여 오염 물질이 유입되었는지를 판단하는 세로 측정 방법을 주로 이용하였다. 그러나 일반적으로 물벼룩은 가로 방향의 움직임이 크기 때문에 카메라를 사용하는 제안된 시스템에서는 그림 1과 같이 수조의 위쪽에서 촬영하는 가로 측정 방법을 이용하여 물벼룩의 움직임을 획득한다.

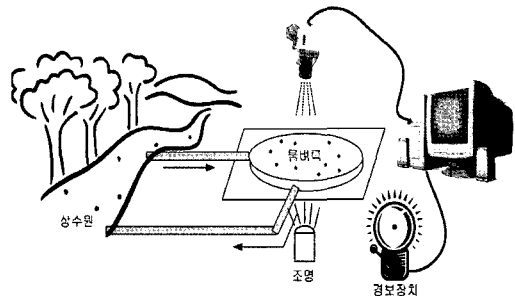


그림 1. 제안된 수질 감시 장치의 구성  
Fig 1. A composition of the proposed water quality monitoring system

카메라를 이용해 영상을 획득할 경우에는 조명 시스템도 함께 고려해야 한다. 일반적으로 카메라를 이용해 획득된 영상은 조명 조건에 민감하게 반응하기 때문이다. 일반적으로 표면을 촬영해야 하는 경우에는 카메라가 있는 방향에서 조명을 조사하는 반사 조명 방법을 이용하고, 물체의 형태를 촬영하는 경우에는 카메라의 반대 방향에서 조명을 조사하는 투과 조명 방법을 이용한다. 제안된 시스템 역시 물벼룩의 표면 보다는 형태와 배치 등을 이용해야 하기 때문에 투과 조명을 사용해 시스템을 구성한다. 투과 조명을 이용하면 배경은 밝고 물벼룩은 어두운 영상을 획득할 수 있다. 조명은 LED를 2차원으로 배열해 전체 수조 영역에 균일한 조명을 비추도록 하며, 이 때 LED 사이에 발생하는 빛의 간섭을 최소화하기 위해서 앞쪽에 확산판 등을 부착하여 사용한다. 그러나 투과 조명을 이용할 경우에는 수조의 뒷면, 또는 아랫면에 조명을 부착해야 하기 때문에 전체적인 시스템의 크기가 커질 수 있다는 단점이 있다.

### III. 객체 추출

연속하는 영상에서 객체를 추출하는 방법 중 가장 보편적인 방법은 연속하는 두 영상 사이에 차영상을 구하는 방법이다. 차영상은 두 영상 사이의 변화를 검출해 내는 방식으로, 두 영상의 같은 위치에 존재하는 화소 값의 차이를 이용해 영상을 구성하는 방법이다. 차영상을 이용하면 비교적 적은 연산으로 두 영상간의 변화를 정확히 검출해 낼 수 있다는 장점이 있다. 본 논문에서 제안하는 알고리즘에서는 연속하는 두 영상 사이의 차영상을 구하는 대신 비모수 예측에 의해 배경 영상을 구성하고 현재 입력 영상과 배경 영상 사이의 차영상을 이용해 객체를 추출해 낸다. 그림 2는 객체를 추출해 내는 과정을 나타낸 순서도이다. 이번 장은 비모수 예측을 이용해 배경 영상을 만드는 방법과 객체의 추출, 인식 오류를 제거하기 위한 잡음 제거 방법에 대해 설명한다.

#### 3.1 배경 영상의 생성

단순 차영상만을 이용하여 배경을 추출해 내고 움직이는 객체를 검출해 내는 방법은 화소값이 카메라 렌즈의 왜곡이나 조명 등의 변화에 민감하게 반응하기 때문에 독립적으로 사용하기 어렵다. 그래서 최근에 차영상을 이용해 객체를 추출해 내는 방

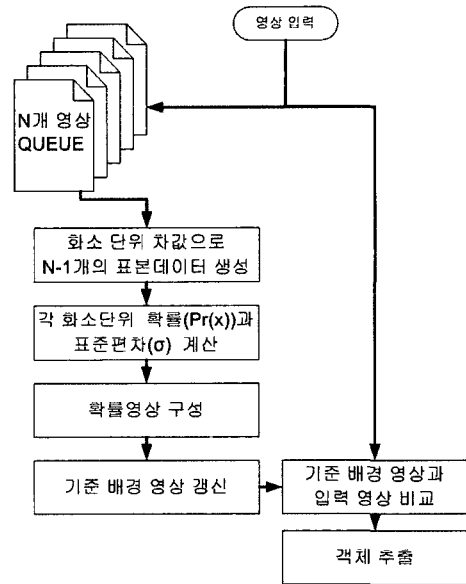


그림 2. 객체 추출 알고리즘의 순서도  
Fig 2. A flow chart for objects segmentation

법은 다른 다양한 방법들과 결합되어 시도되고 있다. 그 중 가장 많이 사용되는 방법은 확률 분포를 이용한 방법이다<sup>5)</sup>. 이는 연속하는 영상에서 동일한 위치에 있는 화소값의 차이로 확률 분포를 구성하고 이 확률 분포를 이용해 객체를 추출해 내는 방법이다. 그러나 확률 분포를 이용해 객체를 추출해 내는 방법은 오인식을 줄이고 정확한 확률 분포를 계산하기 위해서 많은 표본 데이터 영상이 필요하며, 이로 인해 연산이 많아진다는 단점이 있다<sup>6-9)</sup>. 본 논문에서 사용한 비모수 예측 방법은 적은 표본 영상으로 확률 분포를 계산할 수 있다.

비모수 예측을 위해서 우선 확률 분포를 모델링한다. 고정되어 있는 카메라를 통해 획득된 연속하는 영상에서 한 점을 선택해 화소값을 오랜 시간동안 살펴보면 화소값은 객체와 배경을 오가면서 가우시안 분포의 모양으로 변화하는 것을 볼 수 있다. 따라서 우리는 한 화소가 가지는 화소값의 확률 분포를 가우시안 정규분포  $N(\mu, \sigma^2)$ 로 가정한다. 다음으로  $x_t$ 를 시간  $t$ 에서 한 화소의 화소값이라고 하면  $(x_t - x_{t+1})$ 의 확률 분포는 역시 가우시안 분포인  $N(0, 2\sigma^2)$ 가 된다. 이는 같은 위치의 화소값은 동일한 확률 분포를 가지고 있고 연속하는 영상 사이의 시간은 매우 짧으므로 차분은 대부분 0으로 수렴하기 때문이다. 차분에 대한 확률 분포에서 표준 편차를 계산해 내면 정확한 확률 모델을 완성할 수 있다. 표준 편차를 계산하기 위해서 우선 샘플

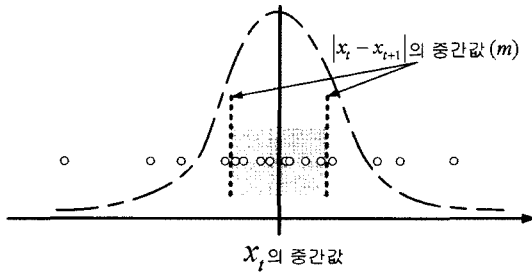


그림 3. 확률 분포와 절대편차중간값의 관계  
Fig 3. A relation between probability distribution and median absolute deviation

영상에서 연속하는 영상의 고정된 한 화소에 대한 차분  $(x_i - x_{i+1})$ 의 절대값을 나열했을 경우 그 중간값을  $m$ 이라고 가정한다. 즉,  $m$ 은 절대편차중간값(MAD, Median Absolute Deviation)을 나타낸다. 차분의 분포는 가우시안 정규 분포이기 때문에 평균을 기준으로 좌우 대칭인 모양이다. 중간값은 샘플들을 크기에 따라서 정렬 하였을 경우 가운데에 위치하는 값이기 때문에 확률 분포에서 중간값은 전체 확률 분포의 50%에 해당한다. 또한  $m$ 은 절대값의 중간값이기 때문에 평균을 기준으로 좌우 25%의 영역이 된다. 따라서 확률 분포에서  $m$ 보다 큰 영역은 전체 확률 분포의 25%에 해당한다. 그림 3은 확률 분포와 중간값  $m$ 의 관계를 나타낸 그림이다.

그림 3에서 원으로 표시된 것들은 화소값  $x_i$ 의 샘플로 16개의 샘플을 예로 들었다. 화소값은 가우시안 정규 분포로 가정했기 때문에 중앙에 더 많은 빈도를 가지고 표시되었다. 그림 3에서 샘플의 중간값은 8번째와 9번째 샘플이 있는 위치가 된다. 이 샘플들로 계산될 수 있는 절대편차중간값인  $m$ 은 전체 확률 분포의 50%에 해당하기 때문에 8샘플을 포함하는 점선 부분이 된다. 따라서 확률 분포에서  $m$ 보다 큰 샘플은 전체 16개의 샘플 중 4개에 해당하고 이는 전체 분포의 25%에 해당한다. 이 과정을 통해 식 (1)을 만들 수 있다.

$$\Pr(N(0,2\sigma^2) > m) = 0.25 \quad (1)$$

차이값의 확률 분포  $N(0,2\sigma^2)$ 에서  $\sqrt{2}\sigma$ 를  $\bar{\sigma}$ 로 치환하면 확률 분포는  $N(0,\bar{\sigma}^2)$ 가 되어 표준 정규 분포로 표현된다. 표준 정규 분포에서 위쪽 25%에 해당하는 값은 표준 정규 확률표에 따라 0.68이 된다. 따라서 식 (1)은 식 (2) 와 같이 표현될 수 있

고, 최종적으로 계산하고자 하는 표준 편차  $\sigma$ 는 식 (3)에 의해서 계산할 수 있다.

$$m = 0.68\bar{\sigma} \quad (2)$$

$$\sigma = \frac{m}{0.68\sqrt{2}} \quad (3)$$

확률 분포를 모델링하기 위한 두 모수인 평균과 표준 편차를 계산했으면, 다음으로 배경 영상을 구성하기 위해서 영상의 각 화소가 배경이 될 확률을 계산해야 한다. N 개의 표본 영상을 기준으로 각각의 화소가 배경이 될 확률은 식 (4)과 같이 계산된다.

$$\Pr(x_i) = \frac{1}{N-1} \sum_{i=0}^{N-1} \frac{1}{\sqrt{2\pi}\sigma} \exp\left\{-\frac{(x_{t-i} - x_{t-i-1})^2}{2\sigma^2}\right\} \quad (4)$$

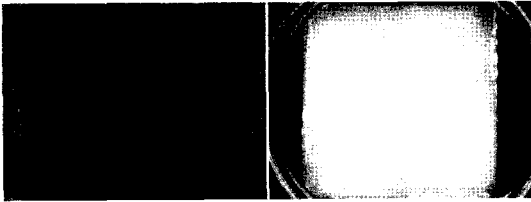
식 (4)에서 N은 표본 데이터 영상의 수이고,  $x_t$ 는 입력 화소 값이다.  $\sigma$ 는 표준편차로 식 (3)를 이용해 비모수 예측 방법으로 계산할 수 있다. 식 (4)를 통해 입력 영상에서 각각의 화소가 배경이 될 확률을 계산하여 확률 영상을 구성한다. 확률 영상을 구성했다면 미리 정해놓은 확률 임계값을 기준으로 배경으로 간주될 화소와 그렇지 않은 화소로 구분한다. 배경으로 간주될 화소는 배경 영상을 갱신하는데 사용된다. 실제 영상 촬영 시에는 렌즈의 왜곡이나 조명의 미세한 변화에 의해서 화소의 그레이레벨 값이 지속적으로 변하게 되기 때문에 매번 새로운 영상이 입력될 때마다 배경을 갱신해 줌으로써 오류가 발생할 확률을 최소화 한다[10]. 식 (5)는 배경을 갱신하는 과정을 설명하고 있다.

$$B(x_t) = \alpha x_t + (1-\alpha)B(x_{t-1}), \text{ if } \Pr(x_t) > th_b \quad (5)$$

B는 배경 영상이고,  $th_b$ 는 배경이 될 확률의 임계값이다.  $\alpha$ 는 입력 영상에서 획득된 배경과 배경 사이에 가중치를 결정해 주는 요소로  $0 \leq \alpha \leq 1$ 의 값을 가진다. 그림 4는 확률 영상과 확률 영상을 기반으로 생성된 배경 영상이다. 배경 영상에서 중앙에 움직임이 없는 객체는 배경으로 인식됨을 볼 수 있다.

### 3.2 객체 추출

배경 영상이 만들어지면 현재 입력 영상과 배경



(a) (b)

그림 4. 확률 분포 영상과 생성된 배경 영상 (a) 확률 분포 영상, (b) 배경 영상  
Fig 4. Probability distribution image and created background image (a) probability distribution image, (b) background image

영상과의 차영상을 구해서 객체를 추출해 낸다. 실험 환경에서 반사 조명이 아닌 투과 조명을 사용했기 때문에 객체 영역은 배경 영역보다 낮은 화소값을 가지게 된다. 따라서 배경 영상과 입력 영상과의 차영상을 구하면 배경 영역은 0~5의 화소값을 가지게 되고 객체 영역은 10~50의 화소값을 가지게 된다. 이렇게 획득된 영상을 7 또는 8을 기준으로 이진화 하면 객체 영역을 분리해 낼 수 있다.

### 3.3 인식 오류를 제거하기 위한 잡음의 제거

3.1절에서 계산한 확률 영상은 전체적으로 잡음 특성을 가지는 화소들이 넓게 분포하고 있음을 알 수 있다. 이런 잡음 특성의 화소들은 배경과 객체를 구분하는데 있어 오인식의 확률을 높이게 된다. 잡음 성분을 줄이기 위해서 영상 분할에서는 EM(Expectation Maximization) 방법 등을 사용한다<sup>[8],[11]</sup>. 그러나 이런 방법들도 역시 많은 연산을 필요로 하는 작업이다. 본 논문에서 제안한 시스템은 실시간 처리를 목적으로 하고 있기 때문에 준수한 성능에 비해 비교적 적은 연산량을 요구하는 형태학적 변형을 사용하여 잡음 성분을 제거했다.

모폴로지(morphology) 기법은 형태학적 변형의 대표적인 방법으로 영상 전반에 걸쳐 발생한 미세한 잡음 성분을 제거하는데 주로 사용된다. 모폴로지 연산에는 침식(erosion), 팽창(dilation), 열림(opening), 닫힘(closing) 연산이 있다. 침식 연산은 물체의 배경을 확장하고 물체의 크기는 축소하는 역할을 하고 팽창 연산은 물체의 최외각 화소를 중심으로 확장하는 역할을 한다. 침식 연산은 물체 주변의 흰 점이나 물체 외각의 두드러진 부분을 제거할 때 사용할 수 있고, 팽창 연산은 반대로 물체 주위의 흰 점이나 물체의 두드러진 부분을 모두 채울 때 사용할 수 있다. 또한 닫힘 연산은 축소와

팽창을 순차적으로 수행하는 것으로, 배경에 생긴 작은 잡음을 제거하기 위해 사용된다. 그러나 객체가 8자와 같은 형태를 가지고 있는 경우 닫힘 연산을 수행하면 한 개의 객체가 두 개의 객체로 분리되는 현상이 발생한다. 따라서 본 논문에서는 닫힘 연산을 수행한 후 열림 연산을 추가로 적용해 줌으로써 객체가 분리되는 현상을 제거 하였다. 모폴로지 기법은 영상 전반에 걸쳐 발생한 미세한 잡음은 제거될 수 있지만 조명의 변화 등으로 생긴 일정 면적을 가지는 잡음 성분은 제거할 수 없다. 따라서 객체로 인식된 영역의 크기와 화소 사이의 상관 관계를 이용한 레이블링 기법을 추가로 적용하였다<sup>[12]</sup>.

레이블링은 하나의 화소를 기준으로 주변에 있는 화소들과의 상관관계를 조사하고 서로 연결되어 있는 화소들을 하나의 영역으로 구분해 내는 과정이다. 레이블링을 거치면 각각의 객체를 독립적인 영역으로 구분해 낼 수 있고, 객체가 가질 수 있는 최소, 최대 영역의 크기를 결정해 줌으로써 작은 크기의 영역을 가지는 잡음을 제거할 수 있는 효과가 있다. 또한 각각의 객체가 구분되기 때문에 각각의 객체가 가지는 영역의 크기와 좌표를 쉽게 찾아낼 수 있다는 장점이 있다.

## IV. 객체 추적

객체 추출 과정을 거쳐 현재 입력 영상에서 다중의 객체를 추출한 후 객체의 추적을 수행한다. 추적 알고리즘은 우선 이전 영상들 사이의 상관관계를 확인하고, 다음 입력될 영상에서 해당 객체가 위치할 수 있는 후보 영역을 설정한다. 새로운 영상이 입력되었을 때 후보 영역에 객체가 발견되면 이전 영상의 어느 객체와 연속선상에 있는지를 판단하는 과정을 수행한다. 이번 장에서는 객체를 추적하기 위해 움직임 정보를 파악하는 방법과, 움직임 정보에 의해 객체를 추적하는 방법, 예외 처리 방법, 경고 발생 조건에 대해 설명한다.

### 4.1 움직임 정보 파악

객체의 구분할 수 있는 정보는 객체의 크기, 위치, 객체가 움직인 속도, 가속도, 방위각 등이다. 최초로 입력된 영상에서는 이런 객체의 정보 중 크기와 위치 정보만을 가질 수 있다. 나머지 정보들은 연속된 두 장의 영상에서 객체들이 정확하게 일치 되도록 만든 후에 계산될 수 있다. 따라서 최초 입력된 두 장의 영상은 크기와 위치 정보만으로 객체

들의 연관성을 조사하게 된다. 움직임이 크지 않은 객체는 이전 영상과 현재 영상 사이에 위치의 변화가 크지 않기 때문에 이전 영상의 객체에 외접하는 사각형을 구하고, 현재 영상의 객체에 외접하는 사각형 영역을 구한다음 두 영역이 서로 겹쳐지는지를 확인한다. 두 영역이 겹쳐지게 되면 두 객체는 동일한 객체라고 판단한다. 이 때 객체의 크기를 추가로 비교하게 된다. 왜냐하면 초당 약 7~10프레임을 처리할 수 있는 실시간 처리이기 때문에 연속하는 두 프레임 사이에서는 크기의 변화가 크지 않기 때문이다.

외접하는 사각형의 겹침에 의해 1차 매칭을 수행하고 난 후 남아있는 객체들은 서로의 거리에 따른 2차 매칭을 수행한다. 남아있는 각각의 객체들의 무게 중심을 기준으로 일대일 비교를 통해 거리를 계산하고, 거리가 정해놓은 임계값보다 작고 가장 짧은 객체들을 서로 매칭 시킨다. 이때도 역시 객체의 크기 정보를 추가로 이용하여 오류 확률을 낮춘다.

비모수 예측을 통해 배경 영상을 만드는 과정에서는 현재 입력된 영상에서 배경 확률을 구하고 이를 기반으로 기존 배경 영상을 갱신하는 과정이 수행된다. 따라서 움직이던 객체가 여러 프레임 동안 멈추어 있다면 해당 객체는 배경으로 인식을 하게 된다. 반대로 초기에는 움직이지 않아서 배경으로 인식되어 있다가 어느 순간 움직이기 시작하는 객체도 존재할 수 있다. 2차 매칭을 끝낸 후에 이전 프레임에 매칭이 되지 않고 남아있는 객체는 움직임이 없어 현재 프레임에서 배경으로 인식된 객체이고, 현재 프레임에 매칭이 되지 않고 남아있는 객체는 움직이지 않고 있다가 어느 순간 움직임을 가지는 객체이다. 전자의 경우는 이전 객체의 정보를 현재 프레임에 그대로 적용시켜 주고, 후자의 경우에는 새로운 객체로 등록시킨다.

모든 객체에 대한 매칭이 끝나면 이전 프레임의 객체들은 현재 프레임의 객체들과 정확히 일대일 매칭이 이루어지게 된다. 일대일 매칭이 이루어진 각각의 객체들 사이에 속도와 가속도, 방위각을 구한다. 여기서 말하는 속도와 가속도는 일반 물리량과는 다른 의미를 가지게 된다. 연속하는 프레임 사이의 시간은 거의 일정하기 때문에 속도는 객체가 이동한 거리를 나타내는 요소가 되고, 가속도는 그 거리가 얼마나 변했는지에 대한 수치로 나타난다. 속도, 가속도, 방위각은 다음 프레임에서 해당 객체가 존재할 위치를 예측하는데 사용된다. 즉, 다음 프레임에서 해당 객체가 발견될 위치는 현재 위치

에서 속도 크기만큼 떨어진 곳이다. 여기에 방위각을 결합시키면 비교적 정확히 해당 객체의 움직임을 예측할 수 있다. 일반적으로 가속도는 속도의 증가량을 나타낸다. 가속도 성분이 양의 값을 가지게 되면 해당 객체는 점점 속도가 높아져 빨리 움직이고 있다는 것을 의미하고, 가속도 성분이 음의 값을 가지게 되면 해당 객체는 서서히 정지하고 있다는 것을 의미한다. 이를 이용해서 속도 성분에 가속도 성분을 더해 줌으로써 최종 위치를 예측한다. 또한 가속도 성분이 큰 경우에는 움직임이 커지기 때문에 객체가 발견될 영역 또한 크게 잡아야 하고, 반대로 가속도 성분이 작은 값을 가지게 되면 작은 움직임을 가지는 객체이므로 객체가 발견될 영역을 작게 잡을 수 있다. 이는 빠른 움직임으로 인한 오차를 최소화 하는 동시에 객체를 찾기 위한 연산을 최소화 시킬 수 있는 방법이다.

$$v = \sqrt{(x_0 - x_1)^2 + (y_0 - y_1)^2} \quad (6)$$

식 (6)은 속도를 계산하기 위해 사용된 기하학적 거리(Euclidean distance)식이다.  $v$ 는 매칭된 두 객체 사이의 속도가 되고  $(x_0, y_0)$ 는 이전 프레임에 위치한 객체의 무게 중심이고,  $(x_1, y_1)$ 은 현재 프레임에 위치한 객체의 무게 중심이다.

방위각은 해당 객체의 움직임이  $x$ 축 양의 방향으로 기준으로 반시계 방향으로 얼마만큼의 각도를 가지는지를 계산한다. 식 (7)은 방위각을 계산하기 위한 식이다.

$$\theta = \tan^{-1} \left( \frac{y_1 - y_2}{x_1 - x_2} \right) \quad (7)$$

가속도 성분은 연속하는 세 장의 영상에 걸쳐 속도의 변화를 가지고 계산한다. 세 장의 영상에서 각각의 객체는 외접하는 사각형의 겹침이나 거리에 따른 매칭을 수행하여 서로 일대일로 대응된다고 가정한다. 가속도 성분은 변화량을 나타내는 것이므로 첫 번째 프레임과 두 번째 프레임 사이의 속도, 두 번째 프레임과 세 번째 프레임 사이의 속도를 각각 구하고 두 성분의 차이값을 계산하면 가속도가 된다.

가속도 성분은 검색 영역의 크기를 결정하는 요소로도 사용된다. 식 (8)은 가속도 성분을 이용해 검색 영역의 크기를 결정하는 식이다.

$$l = a(1 + b^{acc|i|}) \quad (8)$$

$l$ 은 검색 영역의 한 변의 길이를 나타낸다.  $a$ 는 검색 영역의 최소 크기를 나타내는 값으로 본 논문에서는 6을 이용하였다.  $acc[i]$ 는  $i$ 번째 객체의 가속도 값이고 검색 영역은 최소 영역을 기준으로 지수 함수 형태로 증가한다고 가정했다.  $b$ 는 가속도에 따라 검색 영역이 증가하는 양을 결정하는 요소로써 본 논문에서는 실험적 결과에 의해 1.1을 사용하였다. 지수 함수의 형태로 검색 영역의 크기를 변경하면 움직임이 빨라질 경우 발생할 수 있는 오차를 최소화 할 수 있다.

#### 4.2 움직임 정보에 의한 객체 추적

이전 프레임과 현재 프레임 사이에서 속도와 가속도, 방위각 성분이 결정되면 다음 프레임에서 해당 객체는 움직임 정보에 의해 해당 객체를 매칭하는 과정이 수행된다. 우선 움직임 정보에 의해서 한 객체가 다음 프레임에 나타날 수 있는 검색 영역의 중심점을 찾는다. 식 (9)는 검색 영역의 중심점을 찾는 식이다.

$$\begin{aligned} x_i &= x_{i-1} + (v_{i-1} + acc_{i-1})\cos\theta_i \\ y_i &= y_{i-1} + (v_{i-1} + acc_{i-1})\sin\theta_i \end{aligned} \quad (9)$$

$(x_{i-1}, y_{i-1})$ 은 현재 프레임에서 임의의 한 객체에 외접하는 사각형의 무게 중심점이고,  $(x_i, y_i)$ 는 검색 영역의 중심점이 된다. 또한  $v_{i-1}$ 와  $acc_{i-1}$ 는 각각 현재 프레임까지의 움직임 정보를 이용해 계

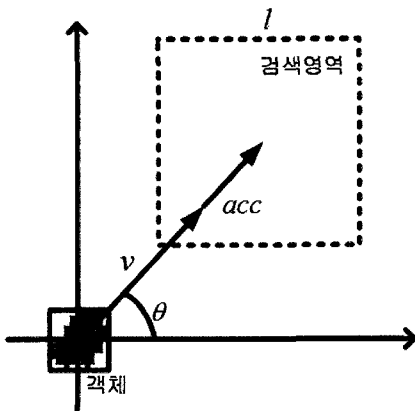


그림 5. 다음 프레임에 객체가 발견될 수 있는 검색 영역을 결정하는 방법  
Fig 5. The method that decide search area which is able to find object in next frame

산한 속도와 가속도 값이고  $\theta_i$ 는 방위각이다. 그림 5는 하나의 객체에서 검색 영역을 결정하는 방법을 나타내었다.  $v$ 는 객체의 속도이고 여기에 가속도  $acc$ 를 더해준다. 움직임 방향은  $x$ 축 양의 방향에서 반시계방향으로  $\theta$ 만큼 회전한 방향이다. 또한 식(8)을 이용해 검색 영역의 크기  $l$ 을 결정한다. 다음 프레임에서 검색 영역 안에 객체가 존재하면 현재 프레임의 해당 객체와 동일한 객체라고 판단한다.

#### 4.3 예외 처리

다중 객체 추적 방법에서 가장 문제가 되는 부분이 두 개 이상의 객체가 서로 모였다가 헤어지는 경우이다. 실제 객체가 이동하는 공간은 3차원인데 반해 촬영된 영상은 2차원이기 때문에 촬영된 영상에서는 깊이 정보를 표현할 수 없다. 따라서 두 개 이상의 객체가 충돌하였을 경우 통과를 하는 경우와 부딪혀 반대 방향으로 이동을 하는 경우인지 구분할 수 없다. 본 논문에서는 두 개 이상의 객체가 충돌하는 경우 해당 객체의 모양 정보를 이용해 이후 이동 방향을 예측하는 방법을 사용하였다. 일반적으로 물벼룩은 원형이 아니라 머리와 꼬리 쪽으로 길쭉한 타원형 모양을 띄고 있다. 따라서 충돌할 경우 물벼룩의 모양을 파악하여 머리 쪽이 향하는 방향으로 진행해 나간다고 가정한다.

또한 물벼룩을 이용한 실험에서 움직이다가 제자리에 멈추는 객체들을 볼 수 있고, 반대로 초기에는 멈추어 있다가 어느 순간부터 움직이기 시작하는 객체들을 볼 수 있다. 이런 객체들에 대한 예외처리를 해줘야 다중 객체 추적이 정확하게 수행될 수 있다. 우선 움직이다가 운동을 멈추는 객체는 비모수 예측에 의한 배경 생성 방법에 의해서 현재 입력에서는 객체가 표시되지 않는다. 따라서 이전 프레임에 객체가 존재 하다가 갑자기 사라지는 경우 최종 객체가 위치했던 자리에 항상 객체가 존재하고 있다고 가정한다. 객체가 움직임을 멈추고 정지하는 경우에는 속도와 가속도가 모두 0에 가까워지게 된다. 기준 배경 영상과 현재 입력 영상의 차영상에서는 객체가 보이지 않지만 실제 입력 영상에는 객체가 존재를 한다. 따라서 속도와 가속도가 모두 0에 근사하게 되면 실제 입력 영상에서 최종 위치를 검색해 객체가 존재하면 해당 객체의 위치를 결정해 준다.

반대로 움직임이 없다가 어느 순간 움직임을 가지는 객체의 경우에는 새로운 객체가 발견된 경우이므로 객체 배열의 가장 마지막에 해당 객체를 추

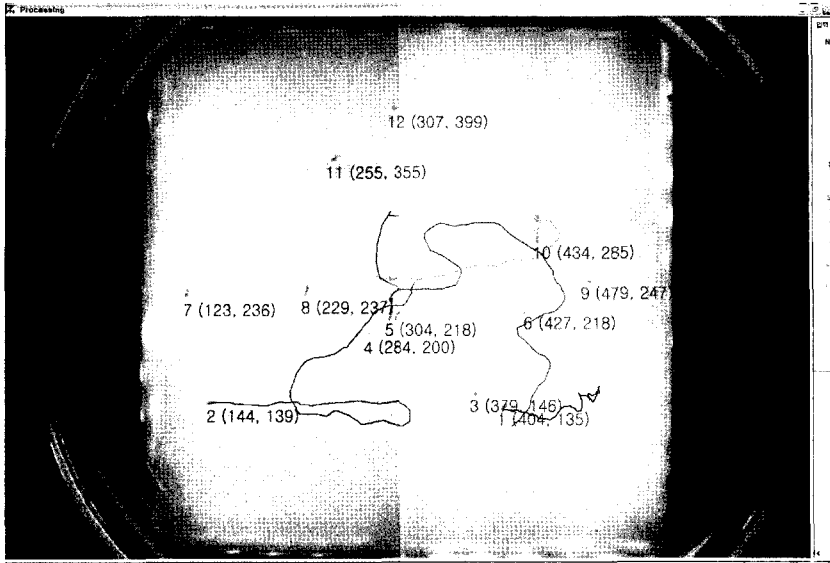


그림 6. 대표 객체의 추적 결과  
Fig 6. Tracking result of representative objects

가해 준다. 이 경우에는 속도와 가속도, 방위각을 구하는 과정이 다시 수행되어야 한다.

#### 4.4 경고 발생 조건

객체 추적을 통해서 각각의 객체들은 움직임 정보를 가지게 된다. 객체 추적이 성공해 객체들 사이에 정확히 일대일 대응이 된다면 해당 객체의 움직임 정보를 계속 추적할 수 있다. 물벼룩이 이상 반응을 나타내는 경우는 우선 움직임이 비정상적으로 빨라지고, 운동 방향이 불규칙하게 된다. 또한 독성이 강한 오염 물질이 유입되었을 경우에는 최종적으로 물벼룩이 모두 죽으면서 움직임이 없는 상황이 발생하게 된다. 따라서 전체 물벼룩의 움직임을 추적하고 있다가 비정상적으로 속도 성분이 증가하거나 속도 성분이 모두 0에 가까워지면 경고를 발생시키게 된다.

### V. 실험 결과

본 논문에서는 실시간 수질감시 시스템을 제안하였다. 실험실의 여건상 고려되어 있는 수조 안에 물벼룩을 넣고 카메라를 이용해 촬영하였다. 실험에 사용한 카메라는 SONY사의 xc-st50 모델을 사용하였고 Matrox사의 Cronos 영상보드를 이용해 640\*480 8bits 그레이 영상을 실시간 획득하였다. 제안된 알고리즘과 소프트웨어는 Microsoft Visual

C++ 6.0을 이용해 구현하였고, 실험에 쓰인 컴퓨터는 3.0GHz로 동작하는 Intel Pentium 4 CPU이다. 그림 8은 전체 알고리즘을 이용해 몇 개의 객체들을 독립적으로 추적한 결과이다. 2번 객체(붉은 선)와 10번 객체(녹색 선)는 이동 과정에서 겹침이 발생한다. 그러나 예외 처리 알고리즘으로 인해 객체가 다시 분리 되었을 때 정확히 개별 객체를 다시 추적함을 볼 수 있다.

제안된 알고리즘을 평가하기 위해 수조에 서로 다른 개수의 물벼룩을 넣고 각각 1000프레임동안 추적 알고리즘을 적용하였다. 추적 성공 횟수는 1000프레임 동안 매칭을 시도한 횟수 중 사람이 육안으로 판단해 결정한 객체의 중심점과 알고리즘에 의해 매칭된 객체의 중심점 사이의 거리가 3 미만일 경우 추적 성공으로 판단하였다. 각각의 실험 횟수별 총 매칭을 시도한 횟수는 프레임 수×객체 수가 되고 총 횟수와 추적 성공 횟수를 이용해 성공률을 계산한다. 또한 전체 1000프레임을 처리하는 시간을 측정하여 실시간 처리가 가능한지를 판단하였다. 표 1은 다양한 조건에서 추적 알고리즘을 적용한 결과이다.

실험 결과 대부분의 실험 환경에서 비교적 정확하게 객체 추적을 수행할 수 있음을 확인하였다. 물벼룩의 객체수가 적을 때 더 좋은 결과가 나온 것은 두 개 이상의 객체가 서로 겹쳐지는 현상이 적게 발생했기 때문에 예외 처리로 인한 오류확률이



표 1. 다양한 조건에서의 추적 알고리즘 결과 비교  
Table 1. The comparison of tracking algorithm in various conditions

실험 횟수	프레임 수	객체 수	추적 성공 횟수	추적 성공률(%)	전체 처리 시간(ms)
1	1000	20	16546	82.7	169500
2	1000	15	13521	90.1	148750
3	1000	12	11326	94.4	141550
4	1000	10	9687	96.9	132190
5	1000	5	4981	99.9	106370

줄어들었기 때문이다. 또한 수조 안에 물벼룩의 객체 수를 적절히 조절할 경우 약 7 frame/sec의 처리시간을 얻을 수 있어 실시간 처리가 가능함을 확인할 수 있었다.

### VI. 결론 및 고찰

본 논문에서는 비전 카메라와 다중 객체 추적 알고리즘을 이용한 실시간 수질 감시 시스템을 제안하였다. 제안된 수질 감시 시스템은 오염 물질이 유입되었을 경우 물벼룩의 행동 변화를 순간적으로 판단하여 최단시간에 위험을 경고할 수 있도록 개발되었다. 비모수 예측을 이용해 빠르고 정확하게 배경과 객체를 구분해 낼 수 있었고, 움직임 정보를 이용한 추적에서는 예외 처리 알고리즘을 적용하여 추적 성공률을 향상시켰다. 실험 결과 객체 추적 성공률은 실험 환경에 사용된 객체 수에 반비례 한다. 또한 객체 수와 데이터의 신뢰성 역시 반비례 관계가 있다. 객체 수를 조절하여 여러 번의 실험을 한 결과 상용화 시스템에서는 약 10~12 마리의 물벼룩을 이용하면 데이터의 신뢰도 측면이나 추적 성공률 측면을 모두 만족시킬 수 있으리라 생각된다.

본 논문에서 제안하는 수질 감시 시스템은 실시간 수질 감시뿐만 아니라 독성 물질이 유입될 경우 물벼룩의 행동 특성을 분석하고 데이터화 하는데 응용될 수 있다. 현재는 사람의 이해 간단한 데이터만을 수집하고 있지만 본 시스템을 이용하게 되면 더 많은 데이터를 획득할 수 있으며, 영상을 직접 저장하면 또 다른 실험 없이 재연이 가능하다는 장점이 있다.

그러나 제안한 시스템에 사용한 추적 알고리즘 역시 여러 개의 객체가 충돌한 후 나타나는 행동을 정확히 예측할 수는 없었다. 경고 시스템에서는 전체 움직임 양에 의해 경고만 발생시키면 되기 때문에 약간의 오차는 인정될 수 있지만, 오염 물질이

유입 되었을 때 물벼룩이 나타내는 움직임을 데이터화하기 위한 시스템에서는 오차를 최소로 줄여야 한다. 추적 알고리즘을 다양한 방법에서 사용하기 위해서는 객체 충돌 후 나타나는 행동을 정확하게 예측할 수 있는 방법들에 대한 연구가 더 필요하다.

### 참 고 문 헌

- [1] 김창수, *유럽의 생물독성정보장치 설치 현황*, 국립환경과학원, 2000.
- [2] 김세화, *한국산 물벼룩의 분포 및 생육 특성*, 국립환경과학원, 2001.
- [3] 이혜진, 신주남, 박민정, 유재정, 양상용, “생물경보장치를 이용한 중금속 생태독성 연구”, *한국상하수도학회 한국물환경학회 2003공동추계학술발표회 논문집*, pp.209-212, 1997.
- [4] BBE Moldaenke, *Annotations and explanations to operating and software*, 2004.
- [5] S. Yalamanchili, W N. Martin and J K. Aggarwal, “Extraction of moving object description via differencing”, *Computer Graph Image Process*, vol. 18, pp. 181-201, 1982.
- [6] N. Friedman and S. Russell, “Image segmentation in video sequences: A probabilistic approach”, In *Proc. UAI*, pp. 165-181, 1997.
- [7] Koller, D., Weber, J., Huang, T., Malik, J., Ogasawara, G., Rao, B., Russell, S., “Towards robust automatic traffic scene analysis in real-time”, *Pattern Recognition, 1994. Vol. 1 - Conference A: Computer Vision & Image Processing., Proceeding of the 12th IAPR International Conference on*, vol.1, pp. 126-131, 1994.
- [8] Srauffer, C., Grimson, W.E.L., “Adaptive background mixture models for real-time tracking”, *Computer Vision and Pattern Recognition, 1999. IEEE Computer Society Conference on.*, vol.2, pp. 246-252, 1999.
- [9] Elgammal, A., Duraiswami, R., Harwood, D., Davis, L.S., “Background and foreground modeling using nonparametric kernel density estimation for visual surveillance”, *Proceedings of the IEEE*, vol.90, no.7pp. 1151-1163, 2002.

- [10] Toyama, K., Krumm, J., Brumitt, B., Meyers, B., "Wallflower: principles and practice of background maintenance", *Computer Vision, 1999. The Proceedings of the Seventh IEEE International Conference on*, vol.1, pp. 255-261, 1999.
- [11] Y. Ren, C. S. Chua, Y. K. Ho, "Motion detection with nonstationary background", *Machine Vision and Application*, vol.13, pp. 323-343, 2003.
- [12] L. G. Shapiro, G. C. Stockman, *Computer Vision*, Prentice Hall, pp. 56-63, 2001.

양 원 근 (Won-Keun Yang)

정회원



2004년 2월 : 인하대학교 전자공  
학과 졸업  
2006년 2월 : 인하대학교 전자공  
학과 석사  
2006년 3월~현재 : 인하대학교  
전자공학과 박사과정  
<관심분야> Pattern Recognition,

Multi-target Tracking, MPEG-7 Visual Descriptor

이 정 호 (Jung-Ho Lee)

정회원

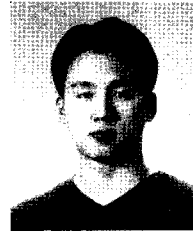


1999년 2월 : 인하대학교 전자공  
학과 졸업  
2001년 2월 : 인하대학교 전자공  
학과 석사  
2007년 2월 : 인하대학교 전자공  
학과 박사  
<관심분야> Multi-view Coding,

Panoramic Image

조 익 환 (Ik-Hwan Cho)

정회원



2000년 2월 : 인하대학교 전자공  
학과 졸업  
2002년 2월 : 인하대학교 전자공  
학과 석사  
2002년 3월~현재 : 인하대학교  
전자공학과 박사과정  
<관심분야> Image Processing,  
MPEG-4, MPEG-7, SVC(Scalable Video Coding),  
Medical Image Processing, Image Watermarking

진 주 경 (Ju-Kyung Jin)

정회원



2003년 2월 : 인하대학교 전자공  
학과 졸업  
2005년 2월 : 인하대학교 전자공  
학과 석사  
2005년 3월~현재 : 인하대학교  
전자공학과 박사과정  
<관심분야> Video Compressing,

IV(Intelligent Vehicle)

정 동 석 (Dong-Seok Jeong)

중신회원



1977년 2월 : 서울대학교 전기공  
학과 졸업  
1985년 : Virginia 주립 공과대학  
전자공학과 공학석사  
1988년 : Virginia 주립 공과대학  
전자공학과 공학박사  
1998년 3월~현재 : 인하대학교

전자공학과 교수

<관심분야> Machine Vision, Image/Video Coding, Image/Video Database, MPEG-7