

움직임 검출의 작업부하 분석

김혜련, 최동휘, 정용화
고려대학교

e-mail:mahakhl@korea.ac.kr, dongwhee@korea.ac.kr, ychungy@korea.ac.kr

Workload Analysis of Change Detection

HaeLyeon Kim, DongWhee Choi, Yongwha Chung
Korea University

요 약

움직임 검출은 비디오 감시 시스템의 작업부하를 줄여주는 주요한 이슈가 되고 있다. 본 논문에서는 다양한 움직임 검출 알고리즘의 작업부하를 분석하고, 움직임 검출의 정확도와 작업부하를 고려한 경우의 최적 알고리즘을 도출한다. 비디오 감시 시스템에서 획득된 실제 데이터를 이용한 실험 결과, 움직임 프레임 비율이 낮은 환경의 비디오 감시 시스템에서는 차 프레임과 GMM을 이용하는 알고리즘이, 움직임 프레임 비율이 높은 환경에서는 GMM만을 이용하는 알고리즘이 정확도와 수행시간을 통합한 성능지수 관점에서 가장 효과적인 움직임 검출 솔루션이 될 수 있음을 확인하였다.

1. 서론

최근 각종 범죄 또는 상황인지를 위한 감시카메라의 사용이 공공시설 뿐 아니라 농촌의 축사나 농작물들에 대한 보안 및 관리에도 많이 사용되고 있다. 대부분의 비디오 감시 시스템은 전송 데이터가 방대하여 인코딩 단에서 전송하는 비디오 데이터에 대한 실시간 처리가 주요 이슈가 되고 있다. 또한 비디오 감시 시스템의 정확도가 보안 및 관리에 직접적인 영향을 미치는 부분이기 때문에 실시간 처리를 만족하는 동시에 정확도를 최적화 하는 것이 중요하다.

비디오 감시 환경에서는 움직임 검출을 통해 전송되는 데이터를 줄임으로써 실시간 처리를 가능하게 한다. 본 논문에서는 높은 신뢰도 있는 데이터 처리를 위하여 세 가지 알고리즘과 이를 다른 방법으로 융합한 두 가지 알고리즘의 수행시간과 정확도를 측정하였다. 또한 수행시간과 정확도의 상관관계를 분석함으로써 비디오 감시 환경을 최적화하기 위한 움직임 검출 알고리즘을 선택한다.

본 논문의 구성은 다음과 같다. 2장에서는 여러 움직임 검출 알고리즘에 대하여 설명한다. 그리고 3장에서는 실험 환경 및 수행시간과 정확도의 상관관계를 분석하고, 4장에서 결론을 맺는다.

2. 움직임 검출

최근에 다양하고 정밀한 움직임 검출 방법들이 발표되고 있다. 본 논문에서는 차 프레임(Frame Difference), 가중평균(Weighted Mean), 가우시안 혼합 분포(Gaussian Mixture Model: GMM)의 기존 세 가지 알고리즘과 차 프레임과 GMM, 가중평균과 GMM의 융합된 두 가지 알고

리즘을 이용하여 수행시간과 정확도의 상관관계를 분석한다.

2.1. 차 프레임

차 프레임 알고리즘은 움직임 검출의 가장 기본적인 방법으로 입력 영상의 현재 프레임과 이전 프레임의 픽셀 차이를 이용한다.

2.2. 가중평균

가중평균 알고리즘은 입력 영상의 현재 프레임(t)과 적응적인 가중평균 프레임($\mu t-1$)에 상대적인 중요성을 주기 위해서 가중 값($\alpha=0.1$)을 부여하여 새로운 가중평균 프레임을 생성한다. 식1은 가중평균을 구하는 식이다.

$$u_t = \alpha y + (1-\alpha)u_{t-1} \quad (\text{식1})$$

그 후 새롭게 구한 가중평균 프레임과 현재 프레임의 차를 이용하여 움직임을 검출한다.

2.3. 가우시안 혼합 모델(GMM)

영상의 각 픽셀 값을 K개의 가우시안을 혼합한 분포로 모델링하여 객체를 검출하는 방법으로 가우시안의 평균과 분산은 매 장면마다 새로 입력되는 값을 학습함으로써 환경의 변화에 적응하기 때문에 배경의 값을 효과적으로 모델링 가능하다.

2.4. 차 프레임과 가우시안 혼합 모델

수행시간이 비교적 빠른 차 프레임 알고리즘으로 움직임 검출 실험 후 정확도를 높이기 위해 움직임이 검출된 프레임들을 대상으로 GMM을 실행하는 방법이다.

2.5. 가중평균과 가우시안 혼합 모델

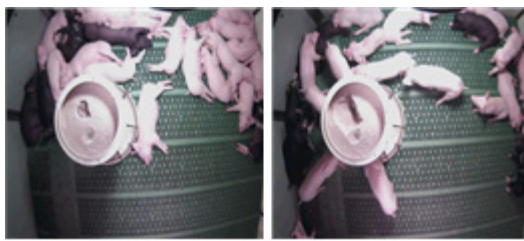
수행시간이 빠른 가중평균 알고리즘 실험 후 GMM을 실행하여 정확도를 높이는 방법이다. 2.4.절과 2.5.절의 알고리즘은 움직임 프레임 비율이 낮은 영상에서 효과적인 것을 실험을 통해 확인할 수 있었다.

3. 실험결과 및 성능분석

3.1. 실험환경 및 수행결과 측정

제안방법의 수행시간 및 정확도 측정을 위하여 노트북 (Intel core i7 M620 2.67GHz 4-코어 프로세서, 4.0GB RAM)에서 30 fps, 640x480 AVI 형식의 5000 frames(2분 46초 분량)으로 이루어진 가축 감시 데이터를 이용하여 움직임 검출 세 가지 알고리즘을 적용하였다.

본 실험에서는 비디오 감시 시스템에서 획득된 실제 데이터를 움직임이 많은 데이터(가축의 이동 및 싸우는 데이터: move.avi)와 움직임이 적은 데이터(잠자는 데이터: sleep.avi)를 이용하여 실험하였다.



(그림 1) 축사영상을 이용한 움직임 검출

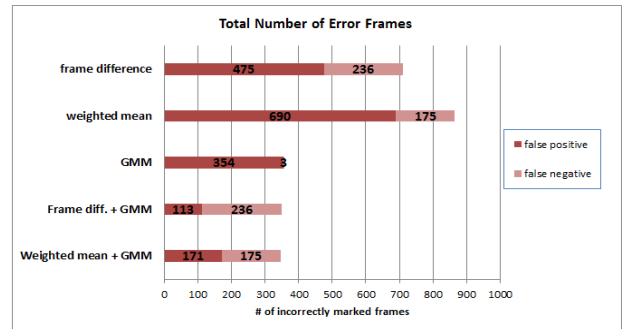
표1은 움직임이 많은 데이터와 적은 데이터에서 수행시간을 분석한 결과이다. 차 프레임과 가중평균 알고리즘은 실시간으로 검출이 가능한 반면, 픽셀 단위 계산으로 작업부하가 많은 GMM은 실시간 검출이 불가능함을 확인할 수 있다. 또한 차 프레임 또는 가중평균 후 GMM을 하는 경우 움직임이 적은 영상에서는 실시간 검출이 가능한 반면, 움직임이 많은 영상에서는 불가능함을 확인할 수 있다.

<표 1> 움직임 검출 수행시간

	sleep.avi	move.avi
Frame Diff.	32.73초	37.20초
Weighted Mean	32.79초	31.85초
GMM	274.99초	281.69초
Frame Diff. + GMM	53.86초	276.98초
Weighted Mean + GMM	65.84초	290.94초

비디오 감시 시스템에서의 오류는 움직임이 검출되지 않아야 하는 상황에서의 움직임 검출(false positive)과 움직임이 검출되어야 하는 상황에서 움직임이 검출되지 않을 때(false negative) 발생하게 된다. 정확도를 분석하기 위해 움직임이 검출되지 않아야 하는 데이터(sleep.avi)와 모든 프레임에서 움직임이 검출 되어야 하는 데이터(move.avi)를 ground truth로 사용하여 각각 false positive와 false negative를 판별하였다.

실험 결과, 정확도 측면에서는 실시간 가능한 알고리즘들이 GMM에 비하여 2배 이상의 많은 오류를 발생시킴을 확인하였다. 그림2는 다섯 알고리즘의 false positive와 false negative 오류 프레임 수를 비교한 것으로 각각 알고리즘의 false positive 프레임 수는 움직임이 검출 되지 않아야 하는 5000프레임 중 움직임이 검출된 오류 프레임 수이고, false negative 프레임 수는 움직임이 검출되어야 하는 5000프레임 중에서 움직임이 검출되지 않은 오류 프레임 수이다.



(그림 2) 총 오류 프레임 수 비교

3.2. 수행시간과 정확도 사이의 상관관계 분석

비디오 감시 시스템의 사용 목적과 환경에 따라 요구되어지는 움직임 검출의 정확도와 수행시간의 중요도는 달라진다. 또한 해당 영상 데이터에서 움직임 프레임이 차지하는 비율에 따라 움직임이 많은 환경과 움직임이 거의 없는 환경으로 나누어진다. 이런 점을 고려하여 본 논문에서는 다섯 가지 알고리즘의 수행시간과 정확도를 측정된 결과를 가지고 여러 알고리즘 중에서 사용자가 요구하는 정확도와 수행시간의 중요도에 가장 적합하고 비디오 감시 환경을 최적화하기 위한 움직임 검출 알고리즘을 선택하기 위해 상관관계를 분석하였다.

<표 2> 수행시간과 정확도 정규화 값

	수행시간		정확도	
	M=0%	M=100%	M=0%	M=100%
Frame Diff	0.12	0.13	0.24	0.01
Weighted Mean	0.12	0.11	0.16	0.02
GMM	1.00	0.97	0.32	1.00
Frame Diff +GMM	0.20	0.95	1.00	0.01
Weighted Mean +GMM	0.24	1.00	0.66	0.02

<표2>는 실험을 통해 측정된 다섯 알고리즘의 수행시간과 정확도를 성능 값 'P'를 구하기 위한 값으로 사용하기 위해 정규화 한 값으로 각각 움직임 비율이 0%, 100% 일 경우의 수행시간은 $Time_{M0}[i]$, $Time_{M1}[i]$ 으로 정의하고 정확도는 $Accuracy_{M0}[i]$, $Accuracy_{M1}[i]$ 으로 정의했다(i는 1~5, 각각의 수는 다섯 알고리즘을 나타낸다).

정확도와 수행시간은 영상 데이터의 움직임 프레임 비율에 따라 적합한 값을 가지게 된다. 움직임 검출을 하기 위한 영상 데이터의 움직임 프레임 비율 'M'은 식2와 같다.

$$M = \text{움직임 프레임 수} / \text{총 프레임 수} \quad (0 \leq M \leq 1) \quad (\text{식2})$$

움직임이 100%인 M1의 값은 움직임이 있는 프레임 비율 M에 영향을 주고 움직임이 0%인 M0의 값은 움직임이 없는 프레임 비율(1-M)에 영향을 주기 때문에 식3과 같이 계산하였다.

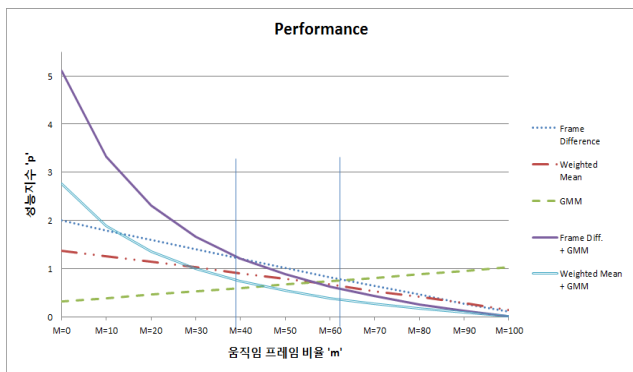
$$Accuracy_M = Accuracy_{M1} \times M + Accuracy_{M0} \times (1 - M)$$

$$Time_M = Time_{M1} \times M + Time_{M0} \times (1 - M) \quad (\text{식3})$$

비디오 시스템 환경에서 요구되는 정확도와 수행시간의 중요도에 따라 성능 지표 'P'값이 계산된다. 식4는 정확도와 수행시간의 중요도에 따른 가중치가 부여된 성능 지표 'P'값을 정의한 식이다. 높은 정확도와 짧은 수행시간을 갖는 알고리즘일수록 효율적인 움직임 검출이 가능하기 때문에 성능 지표 'P'는 클수록 높은 성능을 나타낸다고 정의했다.

$$P(\text{performance}) = \frac{Accuracy \times Weighted Accuracy}{Execution time \times Weighted Time} \quad (\text{식4})$$

$$(0 < Weighted Time \leq 100 \text{ and } 0 \leq Weighted Accuracy \leq 100)$$



(그림 3) 움직임 프레임 비율에 따른 성능 지표

그림3은 요구되는 정확도와 수행시간의 중요도가 같은 경우 영상 데이터의 움직임 프레임 비율에 따른 다섯 알고리즘의 성능을 나타낸 그래프이다.

즉, 위에서 정의된 통합지수 관점으로 평가하면, 움직임 프레임 비율이 약 40% 미만일 경우에는 두 가지 알고리즘을 혼합하여 사용한 Frame Difference+GMM이 가장 낮은 성능을 나타내는 알고리즘에 비해 평균 7배의 높은 성능을 보이고 약 63% 이상의 움직임 프레임 비율이 있을 경우에는 GMM이 가장 낮은 성능을 나타내는 알고리즘에 비해 평균 30배의 높은 성능을 보이는 것을 확인할 수 있다. 따라서 움직임 프레임 비율이 높은 환경의 경우에는 차 프레임과 GMM 알고리즘을 사용, 움직임 프레임 비율이 낮은 환경의 경우에는 GMM 알고리즘을 사용하는 것이 비디오 감시 환경을 최적화하기 위한 움직임 검출 알고리즘인 것을 알 수 있다.

후후 연구로는 실시간 처리와 정확도를 모두 만족시키기 위해 높은 정확도를 나타내는 GMM을 병렬처리 함으로써 실시간 움직임 검출이 가능한 방법을 연구 할 예정이다.

4. 결론

본 논문에서는 의미 있는 움직임만을 모니터링하는 비디오 감시 시스템 환경에 적합한 움직임 검출 알고리즘의 작업부하를 비교 분석하였다. 특히, 각 움직임 검출 알고리즘에서 의미 있는 움직임만을 검출하는데 필요한 작업 부하와 정확도의 상관관계를 분석하였고, 이에 따른 최적의 알고리즘을 결정하였다.

비디오 감시 시스템에서 획득된 실제 데이터를 이용한 실험 결과, 움직임 프레임이 비율이 낮은 환경의 비디오 감시 시스템에서는 차 프레임과 GMM을 이용하는 알고리즘이, 움직임 프레임 비율이 높은 환경에서는 GMM만을 이용하는 알고리즘이 정확도와 수행시간을 통합한 성능지수 관점에서 가장 효과적인 움직임 검출 솔루션이 될 수 있음을 확인하였다.

감사의 글

이 논문은 2012년도 정부(교육과학기술부)의 재원으로 한국연구재단의 지원을 받아 수행된 연구임(No. 2009-0086148)

참고문헌

- [1] F. Nilsson and A. Communications, Intelligent Network Video : Understanding Modern Video Surveillance Systems, Taylor & Francis, 2008.
- [2] N. Paragios and R. Deriche, "Geodesic Active Contours and Level Sets for the Detection and Tracking of Moving Objects," IEEE Tr. on Pattern Analysis, Vol. 22, No. 3, pp. 266-280, 2000.
- [3] O. Javde and M. Automated MultiCamera Surveillance - Algorithms and Practice, Springer, 2008.
- [4] B. Kisacanin, S. Bhattacharyya, and S. Chai, Embedded Computer Vision, Springer, 2009.
- [5] W. Hu, et al., "A Survey on Visual Surveillance of Object Motion and Behaviors," IEEE Tr. Systems, Man, and Cybernetics - Part C, Vol. 34, No. 3, pp. 334-352, 2004.
- [6] M. Patricio, et al., "Computational Intelligence in Visual Sensor Networks: Improving Video Processing Systems," Studies in Computational Intelligence, 96, pp. 351-377, 2008.
- [7] T. Chen, et al., "Computer Vision Workload Analysis: Case Study of Video Surveillance Systems," Intel Technology Journal, Vol. 9, No. 2, pp. 109-118, 2005.