

동적인 배경에서의 사람 검출 알고리즘

People Detection Algorithm in Dynamic Background

최 유 정^a, 이 동 렬^a, 김 윤^{a*}

Yu Jung Choi^a, Dong Ryeol Lee^a, Yoon Kim^{a*}

^a Dept. of Computer and Communication Eng., Kangwon National University, (24341) Room 604, 6th Engineering Building, Kangwon National Univ., Gangwondaehak-gil 1, Chuncheon, Korea

Received 4 October 2018; Revised 19 October 2018; Accepted 24 October 2018

Abstract

Recently, object detection is a critical function for any system that uses computer vision and is widely used in various fields such as video surveillance and self-driving cars. However, the conventional methods can not detect the objects clearly because of the dynamic background change in the beach. In this paper, we propose a new technique to detect humans correctly in the dynamic videos like shores. A new background modeling method that combines spatial GMM (Gaussian Mixture Model) and temporal GMM is proposed to make more correct background image. Also, the proposed method improve the accuracy of people detection by using SVM (Support Vector Machine) to classify people from the objects and KCF (Kernelized Correlation Filter) Tracker to track people continuously in the complicated environment. The experimental result shows that our method can work well for detection and tracking of objects in videos containing dynamic factors and situations.

Keywords: Object Detection, Object Tracking, GMM, Background Cut, SVM, KCF Tracker

1. 서 론

최근 CCTV, 스마트 기기 등의 고성능 기기들이 사용되면서 많은 양의 영상 데이터들이 생성되고, 영상 데이터들을 지능적으로 처리하고 분석하는 영상처리 기술이 비약적으로 발전하고 있다. 특히 취득한 영상에서 객체를 인지하고 필요한 정보를 추출해내는 기술에 대한 관심과 수요가 많다. 이러한 관심은 CCTV감시 보안 시스템, 자율 주행 자동차, 얼굴 인식 등 다양한 분야로 확대되고 있으며, 이에 대한 기본 요소 기술인 객체 검출(Object Detection)기법은 정확성을 높이는 것이 필수적이다.

배경이 고정된 영상에서는 객체의 움직임 영역을 효과적으로 추출할 수 있다. 그러나 동적인 요소들이 포함된 환경에서 객체 검출을 수행하는 것은 매우 어렵다.

예를 들어 바다의 경우에는 파도, 너울과 빛 반사와 같은 동적인 요소들이 포함되어 있다. 그리고 영상에서 날씨, 조도의 변화, 나무의 움직임, 그림자 등의 요소들 때문에 복잡한 환경에서 객체의 검출은 어려운 과제로 되고 있다. 따라서 복잡한 환경에 대하여 강한 객체 검출 알고리즘을 설계하기 위해서는 Fig. 1의 배경모델링과 객체 분류 및 추적 과정에 적합한 방법을 적용할 필요가 있다. 객체를 검출할 때 중요하게 고려해야 할 사항은 Fig. 1의 배경모델링 부분이다. 동영상에서 움직이는 객체를 검출하기 위해서는 배경모델링을 수행하여 배경과 객체를 포함하고 있는 전경을 구분한다.

배경모델링을 구현하는 가장 간단한 방법으로는 Ramesh Jain이 제안한 차 영상을 이용하는 기법이 있다^[1].

* Corresponding author. Tel.: +82-33-250-6387

fax: +82-33-252-6390

E-mail address: yooni@kangwon.ac.kr (Yoon Kim).

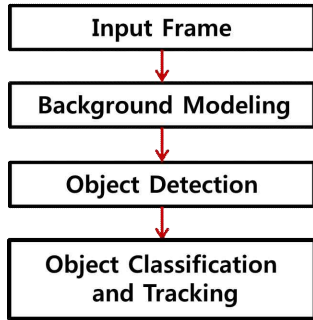


Fig. 1 The general flowchart of object detection and tracking algorithm.

이 기법은 전 프레임과 현재의 프레임의 차이를 이용하여 움직임을 있는 부분을 전경으로 추출하는 방법이다. 그러나 이 방법은 사람뿐만 아니라 동적인 잡음 요소를 모두 추출한다. 따라서 객체와 잡음이 혼합된 전경에서 객체에 대한 정보만을 얻기 위해서는 복잡한 후처리 과정이 필요하다.

이후에는 McFalane이 미디언 필터링(Median Filtering) 배경모델링 방법을 제안하여 객체 검출 성능을 높였다²⁾. 이 방법은 이전의 프레임들을 활용하여 크기가 작은 객체들을 검출하는 방법이다. 이 기법의 기본개념은 이전의 K개의 프레임을 참조하여 좌표마다 픽셀들의 중간 값으로 새로운 배경영상을 만든다. 이 방식의 전경 추출 방법은 매 프레임 갱신된 배경영상과 현재 프레임의 차이를 이용한다. 그러나 미디언 필터를 이용한 방법은 계산 복잡도가 높을 뿐만 아니라 전경 영상에 잔상이 남는다는 단점이 있다. 위의 두 기법은 이전의 프레임 K개의 유사성에 의존하여 하나의 배경 모델을 추정하게 된다. 해당 방법들은 영상에 움직임이 많은 경우에는 정확한 배경을 만들 수가 없고 전경에 잡음이 많이 발생한다는 문제점을 지니고 있다.

위와 같은 문제를 적응적으로 대응하기 위해 Zivkovic은 GMM(Gaussian Mixture Model)을 제안하였다³⁾. GMM은 각 화소에 여러 개의 가우시안 분포를 이용하여 배경 모델을 만드는 통계적인 배경모델링 방법이다. GMM은 프레임이 들어올 때마다 한 화소에 대한 배경 모델을 갱신한다. 미디언 필터링 기법은 하나의 확률 분포만을 사용하지만 GMM은 다수의 확률 분포를 사용하기 때문에 동적인 영상에서도 배경을 강인하게 생성한다. 그러나 움직이는 객체가 한 곳에 일정시간 머무를 때에는 배경 모델을 갱신하는 과정에서 GMM이 객체를 배경으로 인식하는 문제점이 발생한다. 또한, 객체가 다시 움직였을 경우에는 객체가 실제로는 해당 위치에 없지만, 이전의 갱신과정에 영향을 받아 객체의 형상이 배경에 흐릿하게 남는 현상이 발생한다.

Background cut 알고리즘은 동영상에 고정된 배경을 가지고 있다는 가정에 따라 사용할 수 있는 알고리즘이다⁴⁾. 그리고 Background cut은 전경의 경계에 해당하는 픽셀들의 콘트라스트가 배경 픽셀들의 콘트라스트보다 더 높다는 것을 응용하여 만들어졌다. 그래서 이 알고리즘은 컬러 및 콘트라스트 모델을 결합한 기본 모델 덕분에 적은 오류로 배경과 전경을 깔끔하게 추출할 수 있다. 그러나 이 알고리즘은 고정된 배경만을 사용하기 때문에 배경의 변화에 대응하기가 어렵다는 단점이 존재한다.

객체를 검출한 후에 영상에서 객체를 추적하는 방법으로는 평균값 이동(Mean Shift)⁵⁾, 입자필터(Particle filter)를 이용한 방법⁶⁾ 등이 있다. Mean Shift 추적 방법은 윈도우를 이용하여 대상 객체에 대한 색상 히스토그램과 가장 유사한 히스토그램을 갖는 영역을 찾는 방법이다. 그러나 이와 같은 방법은 객체와 배경이 유사한 색상을 가지면 실패할 확률이 높을 뿐만 아니라 객체들의 겹침과 같은 경우에 대응하기 어렵다. Particle filter를 이용한 객체 추적 방법은 객체 내부의 파티클 집합에 대해 시간에 따라 변화한 상태를 확률적으로 추정하여 객체를 추적하는 방법이다. 그러나 이러한 방법은 객체 추적 중에 갑작스러운 조도의 변화나 방향의 변화, 다른 객체에 의한 가림(Occlusion) 현상이 발생할 경우에는 대상 객체를 놓칠 확률이 높아지는 한계가 존재한다. 따라서 위와 같은 상황에 필요한 알고리즘은 객체와 주변 환경을 잘 구분해야 하며 객체들의 겹침, 객체 크기의 변화, 객체의 형태 변화 등의 경우에 강건해야 한다.

본 논문에서는 위의 문제들을 해결하고자 기존의 Background cut 알고리즘을 개선하여 Spatial GMM에 Temporal GMM을 결합한 배경모델링 알고리즘을 제안한다. 그리고 검출된 객체에 보정과정을 거친 후, SVM(Support Vector Machine)⁷⁾과 KCF(Kernelized Correlation Filter) Tracker⁸⁾를 이용하여 객체를 분류하고 추적하는 과정을 소개한다. 또한, 실험 영상으로 해변 영상을 이용한다. 왜냐하면 실제 해변 영상은 파도와 너울과 같은 동적인 요소들이 포함되어 있어서 제안하는 알고리즘을 적용하기에 적합하기 때문이다. 실험 결과를 통해 제안하는 알고리즘이 동적인 해변 환경에서도 사람을 강인하게 인식하고 추적함을 확인하였다.

본 논문의 구성은 다음과 같다. 2절에서는 본 논문에서 제안하는 방법을 설명하고 3절에서는 기존의 배경모델링 방법과 제안하는 배경모델링 방법을 비교한다. 4절에서는 실험 결과 영상에서 강인하게 객체를 추적하는 결과를 보여준다. 마지막으로 5절에서는 결론을 맺는다.

2. 제안하는 방법

2.1 제안하는 알고리즘 전반적 구성

본 연구에서 제안하는 전체적인 알고리즘 순서도는 Fig. 2와 같다. 먼저 전처리과정은 입력 영상에 ROI를 설정한 후 영상을 YCbCr 컬러도메인으로 변환하여 영상의 조도 변화에 대응할 수 있게 한다. 그 다음은 Background cut을 응용한 배경모델링 과정을 수행한다. 이 방법은 Spatial GMM과 Temporal GMM을 결합한 배경모델링 방법으로 기존의 방법들보다 더 명확하게 전경과 배경을 구분한다. 그리고 이렇게 생성된 전경 영상에 모폴로지 연산을 이용하여 잡음을 제거하고 객체 부분을 더 깔끔하게 보정한다. 이후 과정은 보정된 결과 영상을 이용하여 객체의 후보군을 검출하고 후보군에 대해 SVM을 이용하여 검출된 객체들이 사람인지 판단한다. 마지막으로 객체를 추적하는 과정은 KCF Tracker를 이용한다. 이 Tracker는 복잡한 배경에 영향을 받지 않고 강인하게 객체를 추적한다.

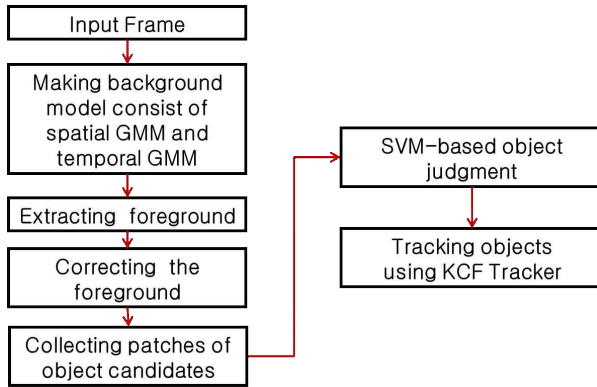


Fig. 2 Flowchart for proposed algorithm

2.2 ROI 설정 및 YCbCr 컬러 도메인 변환

본 절에서는 전처리과정으로 입력 영상에서 사용자에게 좌표들을 입력받아 비정형 ROI(Region of Interest) 마스크를 생성한다. 그래서 해당 비정형 ROI 내에서만 객체 검출 알고리즘을 수행하기 때문에 연산량이 줄어든다. Fig. 3은 ROI 좌표를 설정하는 영상과 ROI 결과 영상을 보여준다. 입력 영상이 RGB 컬러 도메인일 경우에는 조도 변화에 매우 민감하다. 따라서 입력영상은 조도의 변화에 대해 비교적 강인한 특성을 가지는 YCbCr 컬러 도메인으로 변환한다.



Fig. 3 Setting ROI and result images. (a) The setting atypical ROI image, (b) The result image of ROI.

2.3 Gaussian Mixture Model

GMM은 각 픽셀에 여러 개의 각 가우시안 분포를 가지고 있으며 그 분포를 확률밀도함수의 평균과 분산 값으로 모델링을 한다.

$$p(x|X_T, BG + FG) = \sum_{m=1}^M \pi_m N(x, \vec{\mu}_m, \sigma_m^2 I). \quad (1)$$

위의 식은 어느 한 시점 T 에서 한 픽셀의 가우시안 혼합 분포를 나타낸 식이다. x 는 한 픽셀 값, X_T 는 입력 영상, BG 는 배경 영상 가능성의 픽셀, FG 는 전경 영상 가능성의 픽셀이다. 위의 식에서 M 은 한 픽셀이 가지고 있는 가우시안 분포의 개수, π_m 은 m 번째 가우시안 분포의 가중치 값이며 $N(x, \vec{\mu}_m, \sigma_m^2 I)$ 은 평균 $\vec{\mu}_m$ 과 분산 σ_m^2 을 가지는 정규분포를 의미한다. 시점 T 이후의 픽셀 값에 대하여 아래의 식을 사용하여 각 가우시안 분포가 재귀적으로 갱신된다.

$$\pi_m \leftarrow \pi_m + \alpha (o_m^{(t)} - \pi_m). \quad (2)$$

$$\vec{\mu}_m \leftarrow \vec{\mu}_m + o_m^{(t)} (\alpha / \pi_m) \vec{\delta}_m. \quad (3)$$

$$\sigma_m^2 \leftarrow \sigma_m^2 + o_m^{(t)} (\alpha / \pi_m) (\vec{\delta}_m^T \cdot \vec{\delta}_m - \sigma_m^2). \quad (4)$$

위의 식 (4)에서 α 는 갱신 학습 속도, $\vec{\delta}_m^T \cdot \vec{\delta}_m$ 은 공분산을 의미한다. 아래의 식 (5)는 이전 t 시점에서의 해당 픽셀 값이 어떠한 가우시안 분포 분산 내에 포함되는지에 대한 매칭 여부를 나타낸다.

$$o_m^{(t)} = \begin{cases} 1 & , matching \\ 0 & , unmatching \end{cases} \quad (5)$$

가중치 값이 큰 가우시안분포는 T 까지의 시간 내에 다른 가우시안분포보다 더 많은 빈도가 발생하였으므로 배경일 가능성이 높다. 따라서 가중치 값이 큰 순서대로 M 개의 가우시안 분포를 정렬한 후에 입력 픽셀 값이 정렬된 가우시안 분포에 대해 차례로 매칭 여부를 수행한다. 최종적으로 아래의 식 (5)의 조건을 만족하는 가우시안 분포들만이 배경 모델로 이용된다.

$$B = \arg \min_b \left(\sum_{i=1}^b \pi_k > T \right) \quad (6)$$

위의 과정들은 각 픽셀에 대해 위의 식들을 적용하여 동적인 배경 요소에 강인한 배경영상을 도출하고 프레임마다 적응적인 배경 모델을 만든다. 위의 GMM은 한 픽셀에 대하여 시간을 고려한 모델이므로 본 논문에서 Temporal GMM으로 정의한다.

2.4 Spatial GMM과 Temporal GMM을 결합한 Background cut 기반의 배경모델링

Background cut의 기본 개념은 컬러 항과 콘트라스트 항의 합으로 구성된다. 그래서 식 (7)은 두 개의 항을 결합한 깁스 에너지를 표현하였고 이 깁스 에너지를 최소화하는 것이 Background Cut의 목표이다.

$$E(X) = \sum_{r \in \nu} E_1(x_r) + \lambda \sum_{(r,s) \in \epsilon} E_2(x_r, x_s). \quad (7)$$

식 (7)에서 E_1 은 컬러 항을 의미하며 E_2 는 콘트라스트 항을 의미한다. 위의 식에서 x_r 는 픽셀 r 에 대한 레이블 값을 의미하며 X 는 x_r 의 집합을 나타낸다. 또한, 식의 ν 는 한 이미지의 모든 픽셀의 집합을 의미하며 ϵ 은 픽셀 r 의 주변 픽셀 s 들로 이루어진 집합을 의미한다.

컬러 항은 고정된 배경 이미지의 모든 픽셀을 고려한 Global GMM과 픽셀 하나에 대한 가우시안모델을 혼합한 구조로 구성된다. 위의 구성에서 Global GMM은 영상에서 공간에 대한 부분을 고려하므로 Spatial GMM으로 정의하며, 픽셀 하나에 대한 가우시안모델은 픽셀 당 컬러모델로 정의한다.

픽셀 당 컬러모델은 조도의 변화와 배경의 이동에 대해 민감하지만, 객체의 움직임을 검출하는 것은 정확하다. 그러나 Spatial GMM은 픽셀 당 컬러모델보다 정확하지 않지만, 조도의 변화와 배경의 이동에 대해 더 강인하다. 아래의 식 (8)은 컬러 팀의 Spatial GMM을 나타내는 식이다.

$$p(I_r | x = 0) = \sum_{k=1}^{K_b} w_k^b N(I_r | \mu_k^b, \Sigma_k^b). \quad (8)$$

식 (8)의 $p(I_r | x = 0)$ 은 배경의 컬러 항을 의미하며 $x = 1$ 일 경우 전경의 컬러 항을 의미한다. 이때 식 (8)은 미리 설정한 배경 영상인 I_B 를 가지고 모델을 생성한다. 위 식의 $N(I_r | \mu_k^b, \Sigma_k^b)$ 은 가우시안 정규분포를 의미하며 k 는 Spatial GMM의 각 분포의 인덱스를 의미한다.

$$p_B(I_r) = N(I_r | \mu_r^B, \Sigma_r^B). \quad (9)$$

식 (9)는 배경의 픽셀 당 컬러모델을 의미하며 단일 가우시안 분포의 형태이다. 이 식의 μ_r^B, Σ_r^B 은 배경 영상 전체 픽셀의 평균과 분산을 의미한다. 아래의 식 (10)은 위에서 언급한 Spatial GMM의 식 (8)과 픽셀 당 컬러모델의 식 (9)를 혼합한 식이다. 식 (10)의 α 는 Spatial GMM과 픽셀 당 컬러모델을 혼합할 때의 가중치를 나타낸다.

$$p_{mix}(I_r) = \alpha \cdot p(I_r | x = 0) + (1 - \alpha) \cdot p_B(x_r). \quad (10)$$

전경 컬러 모델은 식 (9)에서 나오는 픽셀 당 컬러모델의 배경 확률을 가지고 판단하며 배경의 확률이 매우 낮을 경우에는 전경일 확률이 높게 나타난다. 이를 나타낸 식은 식 (11)과 같다.

$$I_r = \begin{cases} B & p_B(I_r) > t_b \\ F & p_B(I_r) < t_f \\ U & otherwise. \end{cases} \quad (11)$$

위의 식 (11)에서는 B 는 배경, F 는 전경, U 는 불확실한 영역을 의미하며 두 임계 값인 t_b 와 t_f 를 설정하여 전경 컬러모델을 생성한다. 위의 내용을 종합하여 식 (7)의 컬러 항에 대한 식을 아래와 같이 나타낼 수 있다. 식 (12)는 식 (7)의 컬러항 E_1 을 의미하며 이 식에서 x_r 이 0의 값을 가질 경우에는 x_r 이 배경임을 의미하며, x_r 이 1의 값을 가질 경우에는 x_r 이 전경임을 의미한다.

$$E_1(x_r) = \begin{cases} -\log p_{mix}(I_r) & x_r = 0 \\ -\log p(I_r|x_r = 1) & x_r = 1 \end{cases} \quad (12)$$

본 논문에서 제안하는 방법은 위에서 설명한 기존의 Background cut의 컬러 항 부분을 개선한 방법이다. 기존 컬러 항 부분은 Spatial GMM과 픽셀 당 모델로 구성이 되어있는데 이때 사용되는 배경은 시간이 특정 시간에 고정되어있어서 배경의 변화에 대해 대응을 할 수 없다. 따라서 위의 구조에서 시간에 대한 개념을 응용하여 적응적으로 대응할 수 있는 구조를 제안한다. 아래의 그림 Fig. 4는 Spatial GMM만 사용한 경우와 Spatial GMM과 Temporal 모델을 같이 사용한 경우를 비교한 것을 나타낸다. Fig. 4의 (b)의 경우에는 배경에 대한 갱신이 이루어지지 않아 객체와 배경이 혼재되어 전경이 추출되었다. 반면에 (c)의 경우에는 Temporal 모델을 추가하여 영상에 대한 배경이 갱신되어서 깔끔하게 전경이 추출되는 것을 볼 수 있다.

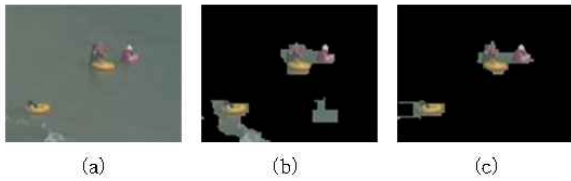


Fig. 4 Results of spatial GMM and spatial GMM with temporal model. (a) Input Image, (b) Result of spatial GMM, (c) Result of spatial GMM with temporal model.

식 (13)은 픽셀 당 모델을 의미하는 식 (9)에 Temporal 모델을 적용한 식이다.

$$p_B(I_{rt}) = \sum_{m=1}^M \pi_{mt} N(I_{rt}|\mu_{mt}, \Sigma_{mt}). \quad (13)$$

식 (13)에서 I_{rt} 는 어느 한 시점 t 에서의 입력 영상을 의미한다. 위의 식은 시간을 고려한 픽셀 당 GMM을 나타낸 식이며 M 은 해당 픽셀의 가우시안 분포의 개수를 의미한다. 제안한 방법은 픽셀 당 모델뿐만 아니라 Spatial GMM 부분에도 이와 같은 Temporal 모델을 적용한다. 또한, Spatial GMM의 경우에는 전체 배경 픽셀값을 고려한 분포를 사용해야 하므로 본 논문에서는 Temporal 모델의 평균값과 가중치 값을 이용하여 배경 픽셀 값을 설정한다. 식 (14)은 기존 Background cut의 고정된 배경을 의미하는 I_B 에 식 (13)의 Temporal 모델을 적용한 것을 표현한 식이다.

$$I_{Bt}(x) = \sum_{x=1}^S \sum_{m=1}^M \pi_{mt} \cdot \mu_{mt}. \quad (14)$$

위의 식에서 x 는 I_{Bt} 의 한 픽셀을 의미하며 S 는 전체 픽셀의 개수를 의미한다. 이때 Temporal 모델에서 각 가우시안 분포의 가중치와 평균값을 곱한 결과를 이용하여 I_{Bt} 를 생성한다. 이렇게 생성된 I_{Bt} 를 이용하여 식 (8)의 w_k^b , $N(I_r|\mu_k^b, \Sigma_k^b)$ 에 적용할 수 있다.

콘트라스트 항은 인접한 두 픽셀이 다른 레이블로 할당될 경우 페널티를 적용하는 부분이다. 두 픽셀의 색이 유사할 경우에는 콘트라스트 항의 값이 증가하며 해당 픽셀이 객체의 경계에 있을 확률이 낮아진다. 아래의 식 (15)는 콘트라스트 항을 나타낸다.

$$E_2(x_r, x_s) = |x_r - x_s| \cdot \exp(-\beta d_{rs}). \quad (15)$$

위의 식에서 d_{rs} 는 L2 norm 식을 적용한 값을 의미한다. 제안하는 방법은 최종적으로 앞에서 수정된 식 E_1 과 E_2 를 합한 식의 결과 값을 최소화하여 객체와 배경을 명확하게 구분을 짓는다. 이때 최소화하는 방법으로 Min cut 알고리즘을 이용한다.

2.5 모폴로지 연산을 이용한 잡음 제거 및 객체 보정

본 절에서는 배경모델링 과정에서 나온 전경 영상을 더 정확하고 깨끗하게 만들기 위해 객체 보정 과정을 진행한다. 먼저, 전경 영상을 이진 영상으로 변환한다. 그 후에 모폴로지(Morphology) 기법의 하나인 닫힘 연산을 수행하여 전경 영상의 잡음을 제거하고 객체의 형태를 보정한다. 이 과정에서 닫힘 연산을 사용한 이유는 객체의 내부에서 소실된 정보를 채우고 전경 영상의 잡음 부분을 제거하기 때문이다.



Fig. 5 The closing operation comparison on before and after images.

- (a) The binary foreground image,
- (b) The image after closing operation.

Fig. 5는 닫힘 연산의 결과를 보여준다. 모폴로지 연산이 모두 끝난 후, 전경 영상에서 객체들을 추출하고 각 객체에 대한 좌표를 저장한다. Fig. 6은 추출된 객체들을 사각형으로 표시한 것이다.



Fig. 6 The image of human object candidates with drawing rectangle.

2.6 SVM을 이용한 사람 객체 판단

객체 보정 과정을 거친 전경 영상에서 추출된 객체 후보군은 파도, 너울, 빛 반사와 같은 동적인 배경 요소들도 포함되어있다. 그러므로 실험 영상에서 사람이 아닌 객체들의 검출을 최소화하기 위해 검출된 후보군에 대한 분류 알고리즘이 필요하다. 따라서 전경에서 추출된 객체들이 사람인지 판단하기 위해 기계학습 기법의 하나인 SVM을 사용한다. 먼저 Histogram of Oriented Gradient(HOG) 특징 기술자를 이용하여 후보군의 특징 벡터를 추출하고 학습 집합을 구성한다^{9, 10}. HOG를 사용한 이유는 영상의 에지 성분을 이용하여 특징 벡터를 생성하므로 객체의 윤곽선으로 객체를 식별할 수 있고 조명에 강인하다는 장점이 있기 때문이다.

SVM은 가능한 최대 마진(Margin)을 가지는 초평면을 찾아 주어진 데이터를 이진 분류하는 기법이다. SVM은 다른 기계학습 알고리즘보다 학습 속도가 빠르고 적은 학습 자료만으로도 분별학습을 수행할 수 있으며 일반화 능력이 우수하다는 장점이 있다. 이때 일반화 능력은 오분류 없이 미지의 패턴을 얼마나 잘 분류하는지에 대한 척도를 의미한다. 두 종류의 데이터를 분류하는 초평면은 아래 식으로 표현된다. w 는 초평면의 법선 벡터이며 초평면의 방향을 나타내고 b 는 초평면의 위치를 나타낸다.

$$d(x) = wx^T + b = 0. \tag{16}$$

모든 샘플을 옳게 분류한다는 조건으로 최대 여백을 갖는 초평면을 찾는 문제는 조건부 최적화 문제라 할 수 있다.

따라서, 이 문제를 해결하기 위해서는 라그랑제 승수를 도입하고 이에 대한 식은 아래와 같다.

$$L(w, b, a) = \frac{1}{2} \|w\|^2 - \sum_{i=1}^N a_i (t_i f(x_i) - 1). \tag{17}$$

식 (17)에서 a 는 라그랑제 승수, t 와 i 는 훈련 집합이다. 그리고 목적함수 $L(w, b, a)$ 를 최대화하기 위해서는 아래 조건을 만족해야 한다.

$$w = \sum_{i=1}^N a_i t_i x_i, \sum_{i=1}^N a_i t_i = 0. \tag{18}$$

식 (18)을 식 (17)에 대입하여 정리한 식은 아래의 식 (19)과 같다.

$$\tilde{L}(a) = \sum_{i=1}^N a_i - \frac{1}{2} \sum_{i=1}^N \sum_{j=1}^N a_i a_j t_i t_j K(x_i, x_j). \tag{19}$$

최종적으로 식 (19)의 매개 변수는 라그랑제 승수 a 만을 갖고 $K(x_i, x_j)$ 는 커널 함수를 의미한다. Fig. 7은 해변 영상에 대한 사람의 이미지와 파도 및 너울의 이미지에 HOG를 적용한 그림이다. 이 그림에서 사람의 HOG는 사람의 형태에 따라 각 셀의 엣지의 방향과 크기가 확실하게 나타나는 것을 볼 수 있다. 그러나 파도 및 너울 이미지의 HOG는 각 셀의 분포가 사람의 HOG보다 매우 다른 방향과 크기를 가지고 있다.

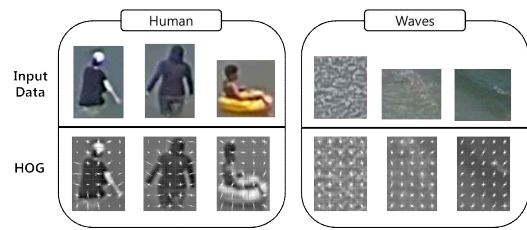


Fig. 7 Input data and HOG descriptor for SVM

따라서 SVM은 위와 같은 HOG 특징 벡터를 갖는 데이터들을 효과적으로 분류할 수 있다. 그리고 비선형 커널을 이용한 학습을 통해 SVM이 다양한 상황에 대응할 수 있도록 한다. 비선형 커널의 종류로는 Radial Basis Function(RBF) 커널을 사용하며 식은 식 (20)과 같다.

$$K(x, y) = e^{-\|x - y\|^2 / 2\sigma^2}. \quad (20)$$

Fig. 8은 샘플 데이터의 학습 과정을 나타낸다. 이때, SVM은 학습의 결과로 특징 공간을 사람인 영역을 긍정(Positive)으로 사람이 아닌 영역을 부정(Negative)으로 나눈다. 이렇게 학습된 SVM을 활용하려면 입력 값은 학습단계처럼 HOG 특징 벡터로 변환해야 한다. 마지막으로 학습된 데이터를 통해 입력 값이 어느 영역에 더 가까운지를 측정하여 해당 객체가 사람에 해당하는지를 판단한다.

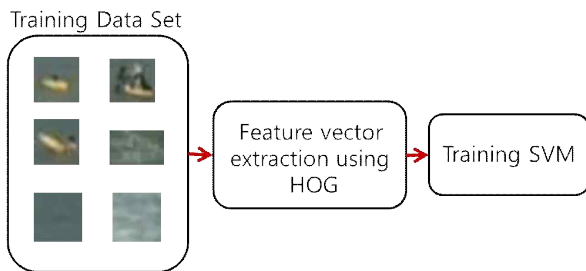


Fig. 8 The proposed process of training data.

2.7 KCF Tracker를 이용한 객체 추적

배경모델링에서 객체의 움직임 변화가 없을 때에는 객체가 배경으로 포함된다는 문제가 있어서 움직임이 적은 객체는 지속적인 검출이 불가능하다. 또한, 영상에서 동적인 배경 요소들이 포함되어 있으면 객체들의 겹침이나 객체의 형태 변화 등과 같은 상황들이 발생한다. 이러한 문제점을 해결하기 위해 KCF Tracker를 이용한다. KCF Tracker는 객체의 겹침과 변형에 대해 다른 Tracker들보다 더 좋은 성능을 보이며 연산 속도가 빠르다는 장점이 있다. 위의 Tracker가 강인하게 객체를 추적하는 요인은 Tracker의 내부 분류기를 학습하는 Online learning을 수행하기 때문이다. 그리고 KCF Tracker는 입력 이미지 샘플 패치에 대해 선형회귀와 비선형회귀를 이용한 모델이다. 또한, 순환 행렬을 이용하여 Tracker 내의 메모리와 계산 복잡도를 많이 감소시킨다. 이때 회귀(Regression)란 기존 데이터들에서 회귀계수를 도출하는 과정인데 이는 실제 데이터로부터 구하지 않은 데이터를 추측할 수 있게 한다. 그리고 KCF Tracker의 주요 특징은 순환 시프트(Circular shift)를 이용하여 순환 행렬(Circular matrix)을 생성한다. 순환 행렬은 Tracker 내부의 분류기를 훈련하기 위해 이용하며 Fig. 9에서 순환 행렬의 구성을 볼 수 있다.

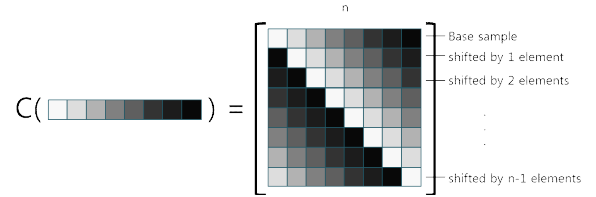


Fig. 9 Circular matrix

식 (22)은 상관관계 필터(correlation filter)를 적용하는 최종 회귀 함수 식을 의미한다. 아래의 식에서 $\hat{f}(z)$ 는 회귀함수, z 는 순환 시프트를 통해 얻은 후보를 의미한다. 다시 말해서 $\hat{f}(z)$ 을 통해 후보 z 를 판단한다. \hat{k}^{xz} 는 x 와 z 의 커널 상관관계를 의미하며 \hat{a} 는 선형 회귀식의 최적화 문제를 해결하기 위해 커널 함수로 대치한 후 DFT로 대각화시킨 필터를 의미한다.

$$\hat{f}(z) = \hat{k}^{xz} \odot \hat{a}. \quad (22)$$

순환 시프트를 통해 객체의 급격한 움직임이나 형태의 변화 등 여러 경우를 고려하므로 Tracker가 동적인 배경요소들의 영향을 크게 받지 않는다. Fig. 10은 Tracker의 내부 분류기 학습에 들어가는 샘플들을 순환 시프트를 하여 추출한 영상이다.

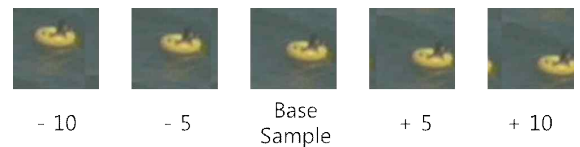


Fig. 10 Samples using circular shift

3. 실험 결과 및 고찰

본 논문에서 제안하는 객체 검출 알고리즘의 성능 평가를 위해서 Fig. 2의 시스템을 구성하여 실험환경을 구축하였다. 1280x720 해상도를 가진 4개의 해변 동영상을 실험 영상으로 사용하였으며 인텔 코어 i7의 컴퓨터로 실험을 진행하였다. Table 1은 4개의 해변 동영상의 정보를 나타낸다. 실험에서 초기 설정 값으로 식 (3)의 GMM의 최대 분포개수는 5개로 설정하였고 α 의 값은 0.002로 설정하였다. 그리고 식 (7)의 λ 의 값은 450으로, 식 (10)의 α 값은 0.5로 설정하였으며, 식 (11)의 문턱치 t_b 값은 0.225σ 로, t_f 값은 1σ 로 설정하여 실험을 진행하였다. 본 알고리즘에서 적정한 단함 연산의 횟수는 1회로 설정하고 실험을 진행하였다.

Table 1 The information of test videos.

video	resolution	channel	the number of the frame
No2	1280x720	RGB	210
No4	1280x720	RGB	200
No6	1280x720	RGB	200
No10	1280x720	RGB	180

Fig. 11은 위의 4개의 영상에 대해 Temporal GMM과 본 논문에서 제안하는 객체 검출 알고리즘 결과를 비교한 것이다. Fig. 11의 (b)는 Temporal GMM만 적용했을 경우이고, (c)는 Temporal GMM을 적용한 후 모폴로지 연산을 수행한 경우이며, (d)는 본 논문에서 제안하는 방법을 수행한 경우이다. (d)를 보면, 제안한 방법이 (b)와 (c)의 결과보다 과도와 같은 동적인 잡음에 대해서 더욱 깔끔하게 진경을 추출한 것을 확인할 수 있다. 그리고 (b)와 (c)를 보면, 정지된 객체의 경우에는 객체의 일부만 추출되거나 잡음이 흩뿌려진 형태로 추출되는 문제점을 가지고 있다. 반면에, (d)에서는 이를 해결하여 객체의 형태가 더욱 완전하고 정확하게 검출하는 것을 알 수 있다.

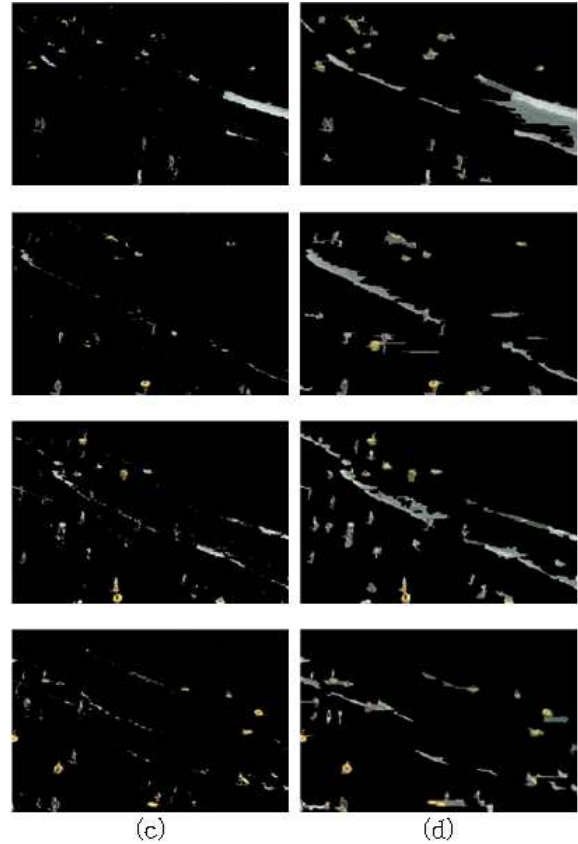
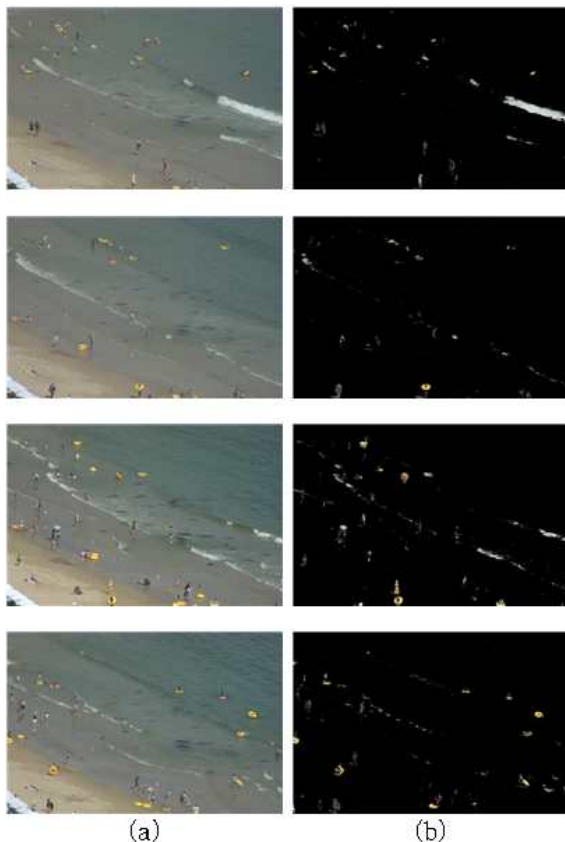


Fig. 11 Experimental results of the proposed background and foreground extraction method; (a) input frame, (b) foreground image of Temporal GMM, (c) foreground image with Temporal GMM and morphology operation, (d) result of proposed algorithm.

Table 2와 Table 3은 기존의 객체 검출 연구들과 제안한 방법의 재현율(Recall)과 정밀도(Precision)를 비교한 표이다. 실험 결과로 아래의 표에서 제안한 방법은 평균 78.13%의 재현율과 평균 74.95%의 정밀도를 보이며 기존의 연구들보다 더 나은 성능을 보인다. 재현율과 정밀도의 식은 아래의 식 (23), (24) 와 같다. 식 (23)의 *Total number of object*는 영상 내의 전체 객체 수를 의미하며 식 (24)의 *Total detections of an algorithm*은 해당 알고리즘을 적용할 경우 검출된 객체 수를 의미한다.

$$Recall(\%) = \frac{\text{detected object}}{\text{Total number of object}} \times 100 \quad (23)$$

$$Precision(\%) = \frac{\text{TRUE detections}}{\text{Total detections of an algorithm}} \times 100 \quad (24)$$

Table 2 The recall of object detection algorithms.

Video	Recall(%)			
	Ramesh Jain [1] + Morphology	McFalan e [2] + Morphology	Zivkovic [3] + Morphology	Proposed
No2	37.0	48.1	58.6	75.8
No4	48.4	58.1	66.6	83.3
No6	55.3	60.5	65.7	84.2
No10	56.4	61.5	64.1	69.2
Average	49.28	57.05	63.75	78.13

Table 3 The precision of object detection algorithms.

Video	Precision(%)			
	Ramesh Jain [1] + Morphology	McFalan e [2] + Morphology	Zivkovic [3] + Morphology	Proposed
No2	42.9	63.6	70.8	78.5
No4	61.5	57.7	62.9	82.7
No6	51.0	47.2	65.1	74.3
No10	52.9	53.3	53.7	64.3
Average	52.08	55.45	63.13	74.95

실험에서 사용된 SVM은 각각 120개의 Positive인 사람 샘플과 Negative인 그 외의 객체 샘플 데이터를 가지고 학습하였다. Table. 4는 SVM을 사용하여 실험 영상에 대한 SVM의 정확도를 나타낸다. 실험 결과는 평균 83.41%의 정확도를 보인다. SVM의 정확도는 식 (25)을 이용한다. 식 (25)의 *Total number of objects*는 검출 알고리즘에서 검출된 객체 후보군의 개수를 의미한다. 그리고 이렇게 학습된 SVM을 이용하여 판단된 입수 자들은 KCF Tracker를 이용하여 추적한다.

$$Accuracy(\%) = \frac{correct\ answer}{Total\ number\ of\ objects} \times 100 \quad (25)$$

Table 4 The accuracy of SVM.

	Video				Average
	No2	No4	No6	No10	
Accuracy(%)	87.0	69.30	97.8	79.47	83.41

Fig. 12는 No2 영상에 제안된 방법을 적용한 결과로 수영경계선을 침범하는 입수 -자에 대한 시물레이션 결과를 보여준다. 하늘색 사각형은 수영경계선을 벗어난 입수자를 나타내며 프레임 수가 증가할수록 입수자들을 더 정확하게 검출하는 것을 확인할 수 있다. 특히 (c), (e), (f)를 보면, 객체의 색상이 주변 배경의 색상과 비슷하여도 제안한 방법이 객체를 놓치지 않고 검출하는 것을 확인할 수 있다. Fig. 13은 No4 영상에 제안된 방법을 적용한 결과이다. (d)를 보면, 뭉쳐있던 입수자들이 서로 멀어지면서 객체의 형태에 대한 변화가 일어나게 되는데, 기존의 연구에서는 이러한 상황에서 객체를 놓치는 프레임이 발생한다. 그러나 제안한 방법을 사용할 경우 분리된 객체를 새롭게 검출할 뿐만 아니라 객체들의 위치를 정확하게 추적할 수 있었다. Fig. 14는 No6 영상에 제안하는 알고리즘을 적용한 결과이며 (a)에서 (e)까지 총 200프레임 동안 바다의 너울이나 파도가 입수자 주변에 계속 존재하는 영상이다. 이러한 상황에서 기존의 연구에서는 너울이나 파도를 전경으로 잡는 프레임이 발생한다. 반면에, 제안한 방법은 해당 문제에 대해 덜 민감하고 두 명의 입수자의 위치를 오랫동안 강건하게 추적한다. Fig. 15는 No10 영상에 적용한 결과이며 입수자의 수가 많아서 객체들의 겹침이 발생하는 영상이다. 객체들의 겹침 및 가림이 발생할 경우에는 제대로 객체를 추적할 수 없는 상황이 발생한다. 그러나 (f)에서 보듯이 제안하는 방법은 트래커의 순환행렬 특성을 이용하여 정확한 객체 추적 결과를 보여주고 있음을 알 수 있다. 최종적으로 실험 결과는 제안하는 방법이 여러 상황에서도 다중 객체가 입수자인지 정확하게 검출하고 추적하는 것을 확인할 수 있다.

그러나 입수자의 크기가 매우 작거나 색상이 주변 배경의 색상과 매우 유사할 경우에는 입수자를 검출하기 어렵다. 또한, 입수자가 같은 위치에서 500프레임 이상 머물 경우에는 Temporal 모델의 갱신 학습 속도와 배경 확률에 대해 영향을 주게 되어서 입수자의 형태가 불완전하게 검출될 수 있다. 그리고 제안하는 방법은 평균 초당 0.03 프레임의 검출 속도를 가지므로 실시간으로 객체를 검출하기에 적합하지 않다.

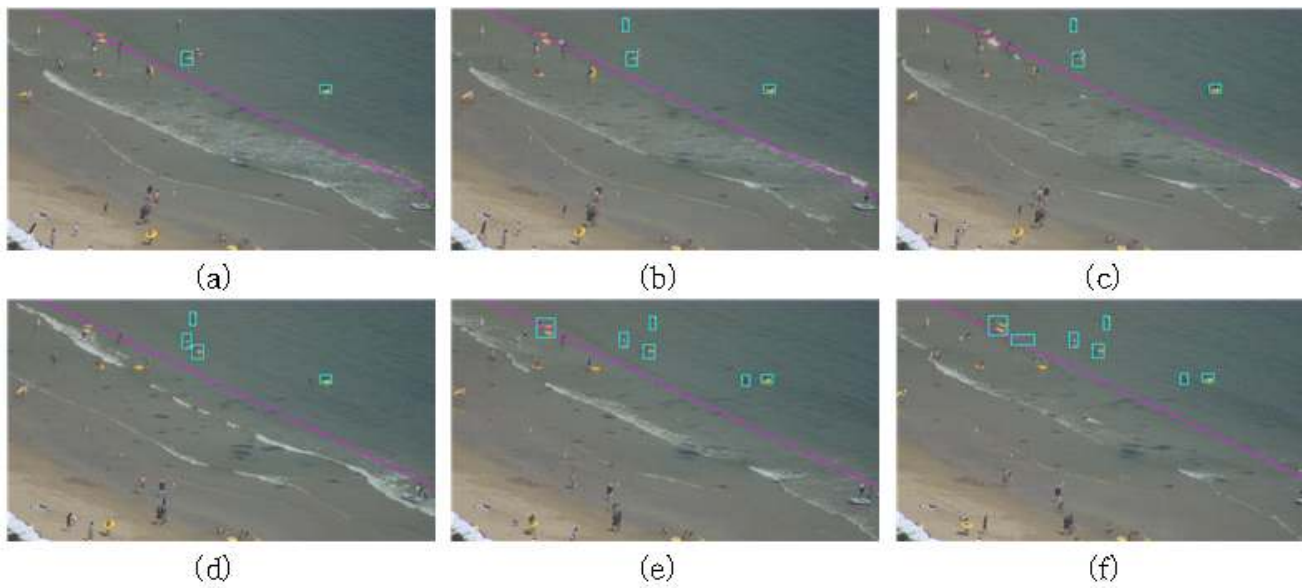


Fig. 12 Experimental results of the proposed method for No2 sequence.(sky blue : tracked object region); (a) 1050th frame, (b) 1072nd frame, (c) 1088th frame, (d) 1141st frame, (e) 1207th frame, (f) 1249th frame.

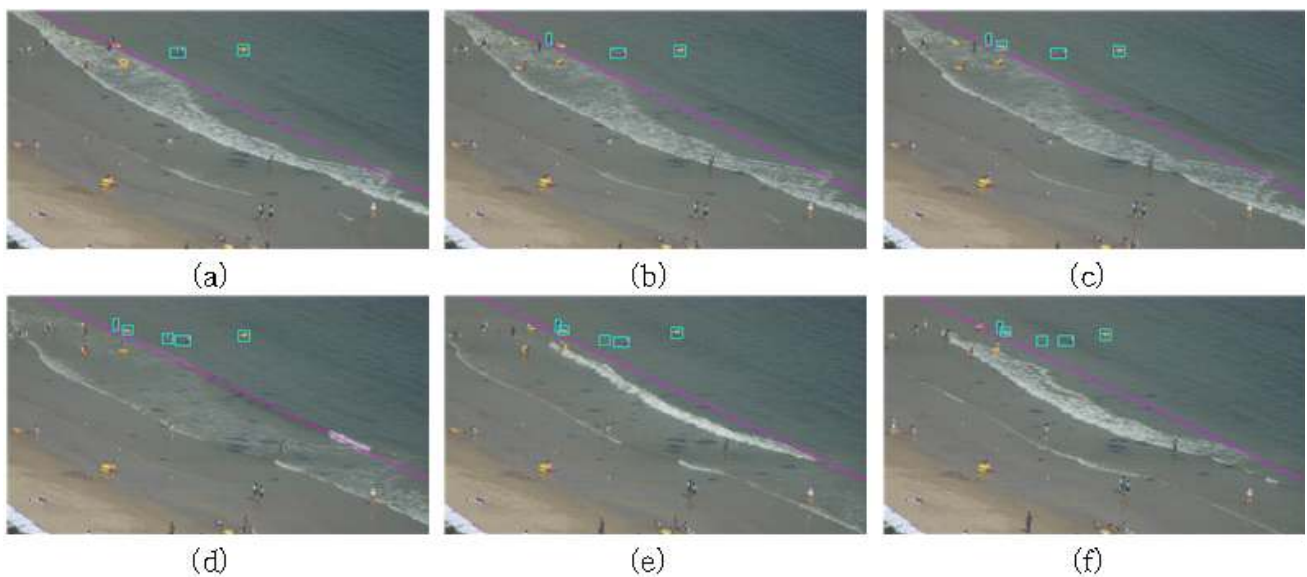


Fig. 13 Experimental results of the proposed method for No4 sequence.(sky blue : tracked object region); (a) 1052nd frame, (b) 1075th frame, (c) 1086th frame, (d) 1130th frame, (e) 1171st frame, (f) 1220th frame.

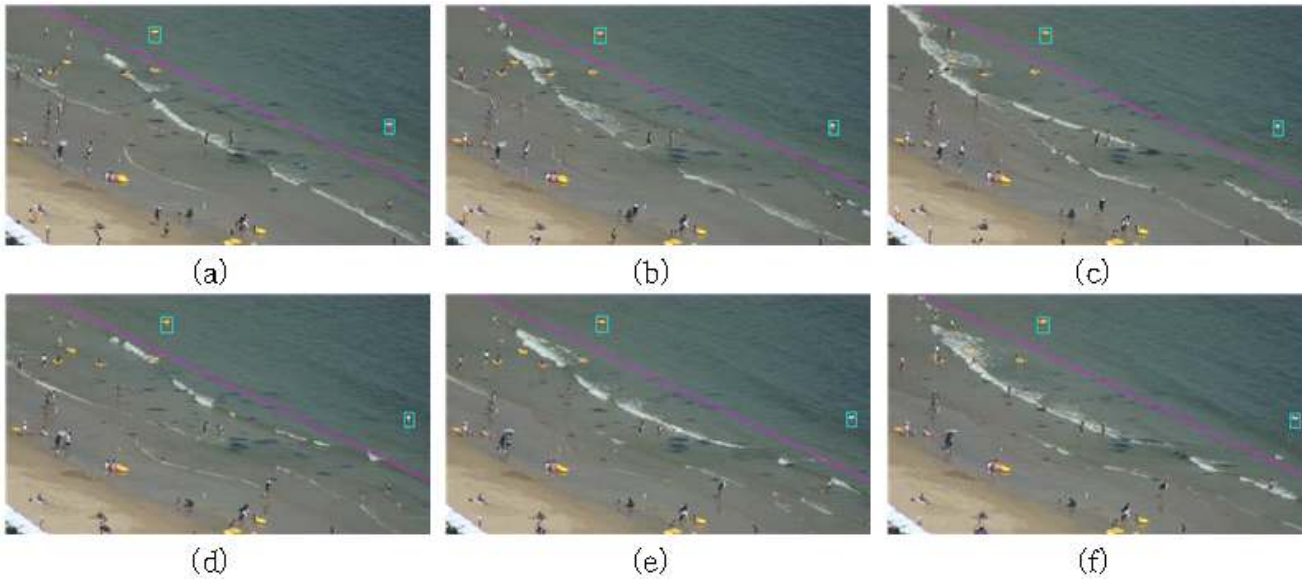


Fig. 14 Experimental results of the proposed method for No6 sequence.(sky blue : tracked object region); (a) 1017th frame, (b) 1052nd frame, (c) 1085th frame, (d) 1155th frame, (e) 1188th frame, (f) 1217th frame.

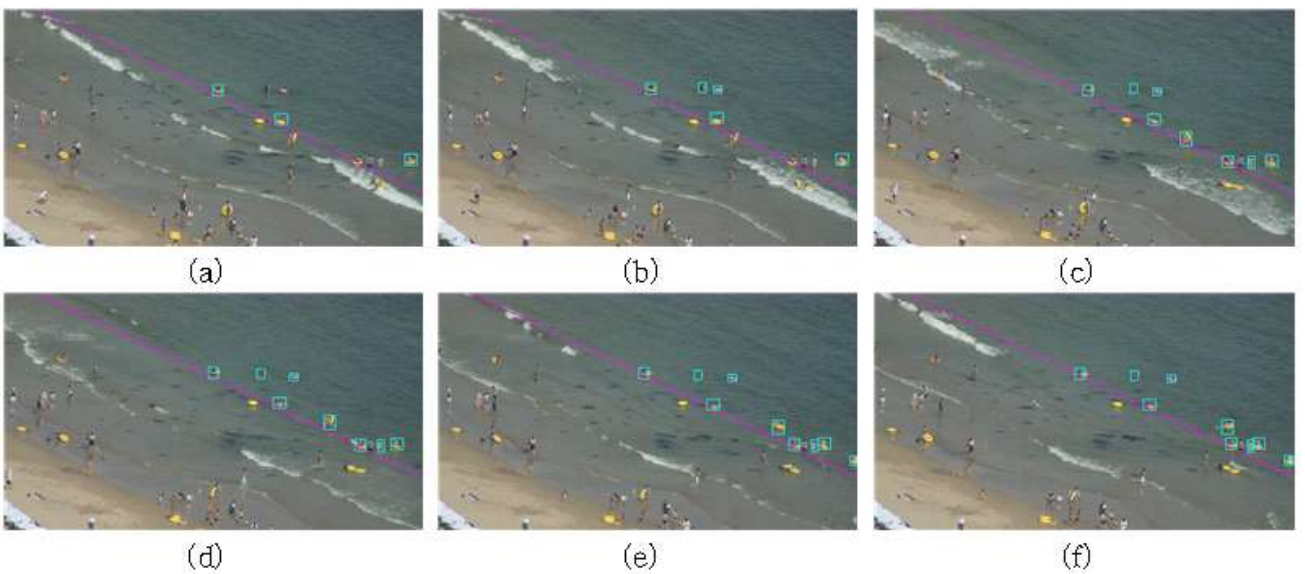


Fig. 15 Experimental results of the proposed method for No10 sequence.(sky blue : tracked object region); (a) 1205th frame, (b) 1226th frame, (c) 1286th frame, (d) 1330th frame, (e) 1377th frame, (f) 1404th frame.

4. 결 론

복잡한 영상에서 조도의 변화, 파도 그리고 물결과 같은 동적인 요소들이 전경 영상에서 잡음처럼 표시되는 문제점이 있다. 또한, 복잡한 영상에서 객체를 추적하기 위해서는 객체와 주변 환경을 잘 구분되어야 하는데 위에서 언급한 동적인 요소들이 객체들의 겹침, 형태 변화 등과 같은 변수에 영향을 미치는 문제점이 있다.

본 논문에서는 이를 해결하고자 복잡한 영상에서 Spatial GMM과 Temporal GMM을 결합한 새로운 배경모델링을 통한 객체 검출 방법과, SVM, KCF Tracker를 사용하여 사람들을 검출하고 추적하는 방법을 제안하였다. 제안한 알고리즘은 해변 영상에서 제안한 배경모델링을 이용하여 기존의 방법들 보다 최대 29%로 향상된 재현율과 22%로 향상된 정밀도를 보여준다. 그리고 이 알고리즘은 SVM을 이용하여 동적인 배경요소를 효과적으로 제거하고 KCF Tracker를 이용하여 검출된 사람들을 복잡한 환경을 가진 영상에서도 강인하게 추적한다. 실험 영상이 해변 영상이라는 특수한 상황에도 불구하고 제안한 방법의 결과는 실험 영상의 다중객체를 검출하며 강건하게 사람들을 추적하는 것을 보였다. 향후 연구로는 객체 검출 모델을 경량화하여 실시간으로 수행할 수 있는 알고리즘을 개발하고자한다. 그리고 일반적인 영상의 복잡한 환경에서 더욱 향상된 검출 성능을 위한 배경 모델링 알고리즘과 분류기를 연구할 예정이다.

References

- [1] R. Jain, 1981, Dynamic scene analysis using pixel-based processes, *Computer*, 14:8 12-18.
- [2] N. J. McFarlane, and P. S. C, 1995, Segmentation and tracking of piglets in images, *Machine vision and applications*, 8:3 187-193.
- [3] Z. Zivkovic, 2004, Improved adaptive Gaussian mixture model for background subtraction, *IEEE Proceedings of the 17th International Conference on Pattern Recognition*, 2 28-31.
- [4] J. Sun, W. Zhang, X. Tang, and H. Y. Shum, 2006, Background cut, *European Conference on Computer Vision*, 628-641.
- [5] I. Leichter, M. Lindenbaum, and E. Rivlin, 2010, Mean shift tracking with multiple reference color histograms, *Computer Vision and Image Understanding*, 144:3 400-408.
- [6] S. Zhou, R. Chellappa, and B. Moghaddam, 2004, Visual Tracking and Recognition Using Appearance-Adaptive Models in Particle Filter, *IEEE Transactions on Image Processing*, 13:11 1491-1506.
- [7] C. J. Burges, 1998, A tutorial on support vector machines for pattern recognition, *Data mining and knowledge discovery*, 2:2 121-167.
- [8] J. F. Henriques, R. Caseiro, P. Martins, and J. Batista, 2015, High-speed tracking with kernelized correlation filters *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 37:3 583-596.
- [9] N. Dalal, and T. Bill, 2005, Histograms of oriented gradients for human detection, *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 1 886-893.
- [10] Q. Zhu, M. C. Yeh, K. T. Cheng, and S. Avidan, 2006, Fast human detection using a cascade of histograms of oriented gradients, *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2 1491-1498.