

論文93-30B-8-6

# 동작 가설 생성과 검증을 통한 이동 물체의 검출 및 추적

## (Detecting and Tracking Nonstationary Objects Through Motion-Hypotheses Generation and Verification)

李眞鎬\*, 崔濬一\*\*

(Jin Ho Lee and Hyung Ill Choi)

### 要約

일정한 시간 간격으로 얻어진 동적 영상을 분석하여 이동 물체를 확인하고 추적하는 작업은 여러 응용 분야에서 사용된다. 본 논문에서는 문제 영역에 대한 사전 지식 및 동작 지식을 이용하여 동적 영상내 이동 물체를 검출 및 추적하는 방법을 제안한다. 즉, 물체 이동에 대한 자취 정보를 이용하여 예상되는 물체 가설(object hypotheses)과 동작 가설(motion hypotheses)을 생성하고 문제 영역에 대한 사전 지식을 이용하여 생성된 가설들을 검증함으로써 제한된 탐색 영역내에서 효율적으로 이동 물체에 대한 검출과 추적 작업을 수행하는 방법을 기술한다.

### Abstract

The tasks which detect and track moving objects, by analyzing dynamic images taken at a constant time interval, are essential in various applications. This paper suggests how to utilize domain-specific knowledge and motional knowledge for detecting and tracking moving objects. That is, The trajectory information of a moving object is to be used for generating hypotheses on expected motion and expected position of moving objects, and the domain-specific knowledge is to be used for verifying the generated hypotheses.

### 1. 서론

동적 영상에 대한 분석은 컴퓨터 비전 분야에서 매우 집중적으로 다루어져 왔다. 일련의 영상들로부터 얻어진 속도 정보는 실세계에서의 3 차원 동작과 구조를 이해하는 데 중요한 역할을 한다. 이러한 속도 정보 및 이동 정보를 추출해 내는 동적 영상 분석 기법에 관한 연구는 공장 자동화, 무인 감시, 교통량 제어, 이동 로봇의 시각 제어, 생의학 분야등 광범위

한 응용 분야에서 그동안 활발히 진행되어 왔다.

Potter<sup>[1]</sup> 와 William<sup>[2]</sup> 은 연속적인 영상을 분석하여 각 화소에 속도 벡터를 할당하고, 할당된 속도 벡터를 이용하여 영상 분할을 수행하는 방법을 제안하였다. Jain<sup>[3]</sup> 은 인접 영상간의 명암차(difference of gray values)를 이용하여 정지 상태의 배경과 이동 물체를 분리하였으며, Chow<sup>[4]</sup> 와 Jian<sup>[5]</sup> 은 영역 정보를 이용하여 후보 영역간의 정합을 시도함으로써 이동 물체를 추적하는 방법을 제안하였다. 그러나 이러한 연구들은 영상에서 얻어진 정보들만을 이용하므로 다음과 같은 문제점들을 내포할 수 있다.

1) 이동 물체의 경계 주위에 자주 발생할 수 있는 흐림(blurring) 현상으로 인해 정확히 이동 물

\*準會員, \*\*正會員, 崇實大學校 電子計算學科

(Dept. of Com. Eng., Sungsil Univ.)

接受日字: 1992年 12月 8日

체의 경계를 추출하기가 어렵다.

- 2) 각 영상에서 추출된 특징들간의 정합을 시도하므로 많은 계산 시간을 필요로 한다.
- 3) 국부적인 특성(local features)들을 이용하므로 정합의 정확성을 기하기 어렵고, 이동 물체간의 폐색(occlusion) 또는 이동 물체와 고정 물체간의 폐색에 대한 처리가 어렵다.

이러한 문제점들을 해결하기 위하여 문제 영역에 관한 지식을 이용하는 방안에 대한 연구도 활발히 진행되고 있다. Yachida<sup>[6]</sup>는 이전 영상에서 얻어진 속도 정보를 이용하여 현 영상에서의 이동 물체의 존재를 예측하는 모델을 제시하고 이를 통해 이동 물체의 검출과 폐색 여부를 판단하는 방법을 제안하였다. 그러나 이 방법은 인접 영상내에서 발생할 수 있는 이동 물체의 구조적인 변화들을 설명하지는 못하였다. Tsotsos<sup>[7]</sup>는 이동 물체가 내포하고 있는 많은 동작 개념들을 다룸으로써 동작의 표현 및 이해를 위한 비교적 일반적인 기본 이론(framework)을 제시하였다. 그러나 한번에 너무 많은 동작 형태를 고려해야 하기 때문에 시스템으로 하여금 이들 중에서 가장 최적의 동작 형태를 선택해야만 하는 부담을 갖게 했다. Frank<sup>[8]</sup>는 세포(cells)추적에 적합한 방법을 제안하였으며 이들은 세포가 갖을 수 있는 상태 변이와 그와 관련된 지식을 규칙 형태로 표현하고, 이를 통해 이동 물체가 일으키게 될 다음 상태를 예측함으로써 연속하는 영상들 속에서 동일 세포의 변화를 추적하는 시스템을 제안하였다.

본 논문에서는 주위 환경에 대한 모델 지식을 구성하고, 이동 물체에 의해 발생 가능한 동작 가설(motion hypotheses)을 지식베이스(rule-base)내 가설 생성 규칙을 이용해 생성함으로써 예측된 제한 영역내에서 이동 물체를 효율적으로 찾아 추적하는 기법을 제안한다. 제안된 기법의 효율성을 확인하기 위하여 사람이 출입문을 열고 들어오는 상황을 고정된 비디오 카메라로 촬영하였으며, 촬영된 영상들을 분석하여 출입문의 움직임과 사람의 움직임을 검출 및 추적하였다. 그림 1은 본 논문에서 제안하는 이동 물체의 검출 및 추적 시스템의 개략적인 구성도를 보여준다.

모델 생성기(model builder)는 참조 영상(reference image)을 입력받아 주위 상황에 대한 사전 지식과 물체에 대한 모델 지식을 형성하는 모듈로서, 형성된 지식은 지식베이스에 프레임(frame)의 형태로 저장된다. 동작가설 생성 및 검증기(motion hypotheses generator and verifier)는 지식베이스내의 물체에 대한 모델 지식 및 동작 가설 생성 규칙과 이전에 얻어진 이동 자취 정보(history of

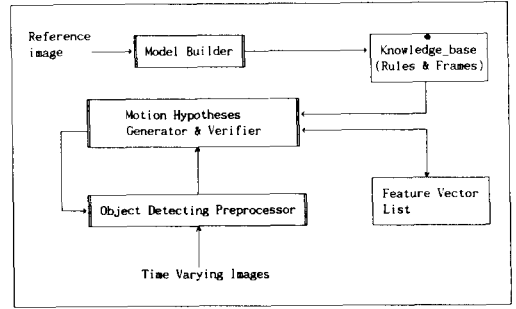


그림 1. 이동 물체의 검출 및 추적 시스템의 구성도

Fig. 1. Schematic diagram of overall system.

trajectory)를 이용하여 이동 물체에 대한 이동 방향과 탐색 영역을 예측함으로써 동작 가설(motion hypotheses)과 물체 가설(object hypotheses)을 효율적으로 생성하고 검증하는 역할을 한다.

물체 검출 전처리기(object detecting pre-processor)는 물체 가설을 검증하는 데 필요로 하는 기본적인 전처리들(예, 영상 분할, 에지 검출)을 수행한다. 특징 벡터 리스트(feature vector list)에는 생성된 동작 및 물체 가설이 검증에 성공했을 때 이들에 관련된 정보와 현 영상에서 얻어진 이동 물체의 위치에 대한 자취 정보가 유지된다. 지식베이스(knowledge-base)에는 물체 및 동작에 관한 지식과 가설 생성 및 검증에 관한 지식이 저장되어 있다. 물체 및 동작에 관한 지식은 참조 영상에 대한 사전 지식 및 장면내의 이동 물체에 관한 지식 그리고 이동 물체들이 가질 수 있는 동작에 관한 지식으로서, 이들은 프레임(frame) 형태로 표현된다. 가설 생성 및 검증에 관한 지식은 규칙(rule)의 형태로 표현되며, 이들은 동작에 관한 지식을 지식베이스로부터 가져와 활성화 시키거나 예상된 동작 가설이 검증에 실패했을 때, 다른 동작 가설을 생성토록 해준다.

## II. 물체 및 동작 지식의 표현(Representation of Object and Motion Knowledge)

본 논문에서는 물체 및 동작에 관한 지식을 가설 생성 및 검증 과정에 적용하여 이동 물체의 검출 및 추적 작업을 수행한다. 따라서, 지식을 어떻게 표현할 것인가하는 문제는 가설 생성 및 검증 과정의 효율성을 결정하는 중요한 요소이다.<sup>[9]</sup> 본 절에서는 지식베이스에 저장될 지식들의 표현법에 대하여 살펴본다. 본 논문에서 사용하는 기본적인 표현법은 프레임

(frame)과 규칙(rule)이다. 프레임은 주위 환경에 대한 모델 지식, 이동 물체에 대한 지식, 그리고 이동 물체의 동작에 관한 지식을 표현하기 위해 사용되고, 규칙은 가설 생성 및 검증 과정을 제어하는 지식을 표현하기 위하여 사용된다.

본 논문에서 사용되는 프레임의 유형은 참조 프레임, 이동 물체 프레임, 그리고 동작 프레임으로 분류될 수 있으며, 각 프레임은 적절한 슬롯(slot)과 슬롯값(slot value)들의 집합으로 구성된다. 참조 프레임(reference frame)은 물체가 이동하는 주위 환경에 대한 정보를 기술하는 프레임으로서, 가설 생성 및 검증기로 하여금 지식베이스내 관련 프레임을 호출할 수 있도록 하는 초기 구동에 사용된다. 즉, 가설 생성 및 검증기는 적절한 시간 간격으로 입력되는 영상을 분석하여 특징값들을 추출하고, 추출된 특징값들에 의하여 참조 영상 프레임의 예시화(instantiation)를 시도한다. 참조 프레임의 예시화가 성공적이지 못한 경우, 즉 프레임의 슬롯값들이 채워지지 못하는 경우, 이는 물체의 이동에 의한 주위 환경의 변화를 의미하며 이동 물체에 대한 검출 및 추적 작업의 개시를 알리는 신호가 된다. 그림 2는 본 논문의 실험에서 사용된 참조 프레임의 기본 구조를 보여준다. 참조 프레임의 구조 및 각 슬롯값의 범위(constraint)는 모델 생성기에 의해 행해진 참조 영상에 대한 분석을 통해 얻어진다.

```
Frame <name of reference frames> With
INFORMATION FOR WALL_REGION : [feature values with constraints]
INFORMATION FOR DOOR_REGION : [feature values with constraints]
end frame
```

그림 2. 참조 프레임의 구조  
Fig. 2. Basic structure of reference frame.

물체 프레임(object frame)은 동작의 주체가 되는 이동 물체를 정의하는 프레임으로서, 이 프레임의 속성값들은 상위 레벨인 동작 프레임으로부터 상속받은 제한된 탐색 영역(CLUE\_SEARCH\_REGION)에 대해 전처리기를 적용함으로써 결정된다. 또한 예시화된 물체 프레임은 상위 레벨의 동작 프레임에 전파되어 동작 프레임의 <Relation> 슬롯값을 결정하기 위하여 사용된다. 그림 3은 이동 물체에 대한 물체 프레임의 기본 구조를 보여준다.

그림 3에서 <Part\_of> 슬롯은 상위 레벨의 동작 프레임과의 연결을 나타내는 슬롯으로서, 슬롯값은 동작 프레임의 이름(즉, OPEN, CLOSE, STOP)이 된다. <Preprocessor> 슬롯에는 절차적 지식(procedural knowledge)으로서 해당 전처리기의 이름이

명시된다. 따라서 영상내 이동 물체에 대한 물체 가설은 활성화된 프레임의 Preprocessor 슬롯에 있는 전처리기를 상위 레벨의 동작 프레임에서 상속받은 CLUE\_SEARCH\_REGION에 적용할 때 검증된다. FEATURE\_INFORMATION 슬롯은 전처리기를 구동하여 얻어진 물체 가설에 대한 특징 정보들을 등록하는 슬롯으로서, 각 특징 정보가 취할 수 있는 값의 범위는 적절히 제한되어 있다. 예를 들어, 이동 물체가 출입문인 경우 FEATURE\_INFORMATION 슬롯은 <Length, Edge\_direction, Approximate\_position>등과 같은 슬롯들을 포함하게 된다.

```
Frame <name of object frames> with
Part_of : [name of motion frames]
Preprocessor : [name of preprocessors]
FEATURE_INFORMATION : [feature values with constraints]
end frame
```

그림 3. 물체 프레임의 구조  
Fig. 3. Basic structure of object frame.

본 논문에서 정의하고 있는 동작 개념은 일반적으로 실세계에서 발생할 수 있는 이동 물체의 동작 상태를 자연어로 표현한 개념을 의미한다.<sup>[7]</sup> 예를 들어, 출입문이 실세계에서 갖게 되는 동작 상태는 '문이 열린다', '문이 닫힌다' 등의 자연어에 의하여 표현될 수 있다. 따라서 입력 영상에 나타나는 출입문의 동작 상태는 OPEN, CLOSE, STOP 등과 같은 동작 프레임에 의하여 정의될 수 있다. 각 **동작 프레임(motion frame)**은 그림 4와 같이 동작의 주체인 물체를 나타내는 슬롯과 동작 상태에 따른 이동 물체의 존재 가능 영역을 예측하는 슬롯, 그리고 확인된 현 영상에서의 이동 물체의 위치와 전 영상에서의 이동 물체의 위치를 비교하여 동작 상태의 적합성을 검증하는 슬롯으로 구성된다.

동작 프레임 구조에서 *Object* 슬롯은 이동 물체 프레임의 이름을 그 값으로 갖는다. 따라서, 물체 프레임은 동작 프레임의 하위 레벨 프레임으로 정의되며, 동작 프레임의 속성은 물체 프레임에 상속된다. 이와 같은 동작 프레임과 물체 프레임 사이의 관계는 그림 5와 같은 기본적인 계층 구조로 표현될 수 있다.

```
Frame <name of motion frames> with
Object : [name of object frames]
CLUE_SEARCH_REGION
Begin : Position of Object at Previous Instance(POPI)
End : POPI ± previous_velocity × time_interval
Relation : [logical functions]
end frame
```

그림 4. 동작 프레임의 구조  
Fig. 4. Basic structure of motion frame.

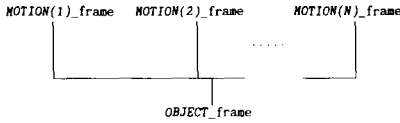


그림 5. 프레임간의 계층 구조

Fig. 5. Basic hierarchical structure of frame system.

그림 4의 CLUE\_SEARCH\_REGION 슬롯은 이동 물체의 존재가 예상되는 영역을 정의하는 슬롯으로서, 이 슬롯의 *Begin*과 *End* 속성값은 물체의 이동 자취 정보(history of trajectory)에 의하여 결정된다. 즉, 전 영상에서의 이동 물체의 위치와 이동 물체의 동작 상태 그리고 영상 촬영 시간 간격을 고려하여 현 영상에서 이동 물체가 존재하리라 예상되는 영역을 결정한다. 이처럼 구해진 예상 영역에 대한 정보는 하위 레벨 프레임인 물체 프레임에 전달되어 물체 프레임의 예시화 작업에 사용된다.

그림 4의 *Relation* 슬롯은 논리적 함수를 호출하여 그 진위값을 확인하는 함수로서, 전 영상에서 예시화된 물체 프레임과 현 영상에서 예시화된 물체 프레임간의 공간적인 관계(spatial relation)를 확인하기 위하여 사용된다. 예를 들어, 물체가 왼쪽으로 이동하고 있는 동작 상태에서는 전 영상에서 예시화된 물체 OBJ<sub>t</sub>과 현 영상에서 예시화된 물체 OBJ<sub>t+1</sub>와의 공간적 관계를 다음과 같이 확인할 수 있다.

$$Relation : LEFT(OBJ_t, OBJ_{t+1})$$

### Ⅲ. 가설 생성 및 검증(Hypotheses Generation and Verification)

이동 물체의 검출 및 추적 작업은 반복적인 가설 생성과 검증 작업에 의하여 구현된다. **가설 생성** (hypotheses generation)이란 지식베이스에 저장되어 있는 프레임들의 집합에서 적절한 프레임을 선택하는 작업(프레임을 활성화 하는 작업)을 의미하고, **가설 검증** (hypotheses verification)이란 선택된 프레임을 입력 영상에 적용하여 예시화시키는 작업을 의미한다.<sup>[9] [11]</sup> 그림 6은 가설을 생성하고 검증하는 과정에 대한 개략적인 구성도이다. 본 논문에서 채택하고 있는 가설 생성 및 검증 방법은 기본적으로 목표 지향적(goal-driven)인 하향식 제어(top-down

control) 방식이다.<sup>[10] [12]</sup> 즉, 현 영상에서 예상되는 동작 상태에 대한 동작 프레임을 호출하여 동작 가설을 생성하고, 생성된 동작 가설은 동작의 주체가 되는 물체 가설을 생성하게 된다.

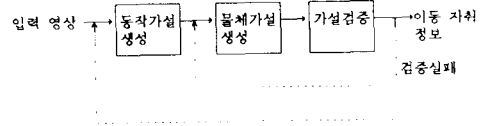


그림 6. 동작.물체 가설 생성 및 검증 과정

Fig. 6. Schematic diagram for generating and verifying hypotheses.

생성된 가설들에 대한 검증 작업(프레임의 예시화 작업)은 물체 가설에 대한 검증부터 행해진다. 물체 가설에 대한 검증 작업을 수행하는 전처리기는 생성된 동작 가설에서 전달받은 정보를 활용하여 작업을 수행하며, 작업 결과(물체 가설에 대한 검증 결과)는 상위 레벨인 동작 가설에 전파되어 생성된 동작 가설의 적합성을 검증하는데 사용된다.

본 논문에서는 적절한 동작 가설의 생성을 위하여, 즉 적절한 동작 프레임의 활성화를 위하여, 지식베이스내의 가설 생성 규칙들을 사용한다. 각 생성 규칙은 전제부와 결론부로 구성되며, 규칙의 전제부에서는 이전 가설의 검증 결과를 분석하고 결론부에서는 분석된 결과에 의하여 다음에 생성될 동작 가설을 결정한다. 동작 가설의 생성을 위한 기본 원칙은 동작의 연속성이다. 즉, 영상을 촬영하는 시간 간격은 매우 작다는 전제하에서 '물체의 동작 상태는 급격한 변화를 하지 않는다'라는 원칙이다.<sup>[14]</sup> 그림 7은 동작 가설 생성 규칙의 기본 구조를 보여준다.

동작 가설 생성 규칙에 의하여 활성화된 동작 프레임의 예시화를 위하여는 물체 이동에 대한 자취 정보(history of trajectory)가 활용된다. 즉, 자취 정보가 동작 프레임의 CLUE\_SEARCH\_REGION 슬롯의 값을 결정하기 위하여 사용된다. 자취 정보에 의하여 이 슬롯의 값이 결정되면 물체 가설의 생성 및 검증 작업을 수행한 후, 동작 프레임의 *Relation* 슬롯에 대한 예시화를 진행한다.

물체 프레임의 활성화 작업은 동작 프레임의 *Object* 슬롯에 등록된 상수값(물체 프레임의 이름)에 의하여 수행되며, 활성화된 프레임의 예시화 작업은 CLUE\_SEARCH\_REGION 슬롯에서 전달받은 정보와 전처리기를 이용하여 수행된다. 물체 프레임의

예시화란 입력 영상에서 대응되는 물체를 확인하는 과정으로서, 전처리기가 적절한 영상 처리 연산자를 적용하여 슬롯값들을 결정한다. 전처리기를 구동하여 추출된 특징들이 활성화된 프레임의 슬롯들을 성공적으로 채울 때 물체 가설에 대한 검증은 완료되고, 이어서 상위 레벨인 동작 가설의 Relation 슬롯에 대한 확인 작업이 시작된다. 따라서 본 논문에서 채택하고 있는 영상 처리는 목표 지향적인(goal-driven) 방식이다. 즉, 물체를 확인하는 과정에 사용될 영상 처리 연산자의 형태 및 적용 범위는 프레임에 기술된 지식에 의하여 결정된다.

```

IF (previous activated frame = MOTION1) &&
   (its instantiation = successful)
THEN (activate MOTION1 frame)

IF (previous activated frame = MOTION1) &&
   (its instantiation = unsuccessful)
THEN (activate MOTION2 frame)
    
```

그림 7. 동작 가설 생성 규칙의 기본 구조

Fig. 7. Basic structure of rules for generating motion hypotheses.

본 논문의 전처리기는 Marr와 Hildreth가 제안한 에지 검출기<sup>[13]</sup>와 검출된 에지의 방향 성분을 확인하는 연산자, 히스토그램(histogram)을 분석하여 임계 작업(thresholding)을 수행하는 연산자, 영역 레이블링(labeling)을 수행하는 연산자, 그리고 각 영역의 크기 및 위치를 계산하는 연산자 등으로 구성되어 있다. 특히, 히스토그램을 분석하여 임계 작업을 수행하는 연산자는 임계값(threshold value)을 자동으로 결정하도록 설계되었다. 예를 들어, 그림 8과 같은 히스토그램에서 두 개의 마루(peak)를 분리하는 골(valley)을 찾기 위하여 마루 크기(peakness)와 골 크기(valleyness)를 이용하여 임계값  $Z_k$ 를 자동으로 결정한다.

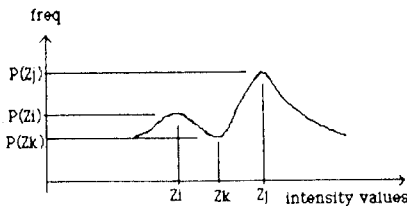


그림 8. 마루크기와 골크기를 이용한 임계값 결정  
Fig. 8. Threshold selection using peakness and valleyness.

- (1) 적어도  $T_{adj}$ 만큼의 거리가 떨어진 2개의 국부적인 최대치(local maxima)를 찾는다. ( $Z_i$ 와  $Z_j$ 에서 생긴다고 가정한다.)
- (2)  $Z_i$ 와  $Z_j$ 사이의 골(가장 낮은점)을 찾는다. (확인된 골을  $Z_k$ 라 하자)

$$(3) \text{ 만일, } \frac{\min(P(Z_i), P(Z_j))}{P(Z_k)} > T_{peak} \text{ 이면 } Z_k \text{를 임계값으로 정한다.}$$

위에서 상수  $T_{adj}$ 은 서로 다른 두 물체의 명암값은 어느 정도 이상의 차이를 가져야 함을 암시하고,  $T_{peak}$ 은 마루 크기와 골 크기의 비율이 일정 크기 이상인 경우만을 고려하기 위하여 사용된다. 한편, 본 논문에서는 히스토그램이 완만(smooth)하지않아 마루와 골을 찾기 어려운 경우를 고려해 제한된 예측 영역으로부터 얻어진 명암 히스토그램을 평활화(smoothing)하는 작업을 선행한다. 이처럼 결정된 임계값은 영역 분할을 수행하여 이동 물체에 대한 후보 영역을 확인하기 위하여 사용될 수 있다.

본 논문에서 채택한 가설 생성 및 검증 방법은 완전한 히향식 제어 방식이다. 즉, 하나의 동작 가설에 대한 검증이 완료된 후 다른 동작 가설을 생성하게 되고, 물체 가설은 동작 가설에 의존하여 생성된다. 이러한 제어 방식은 시스템의 융통성을 위축시키는 단점은 있으나, 처리 과정이 단순하여 시스템의 효율성을 높이는 효과를 갖는다. 특히, 물체 및 동작 프레임에 대한 정의가 주어진 문제에 적합하게 이루어진 경우에는 매우 신속히 이동 물체에 대한 검출 및 추적 작업이 수행될 수 있다. 또한 가설 생성 규칙을 적절히 변화하면, 즉 규칙의 전제부에서 이전 가설의 검증에 대한 다양한 분석을 시도하면, 복잡한 상황에서도 적응성이 높은 시스템이 구현될 수 있다.

#### IV. 실험 및 결과(Experiments and Results)

본 논문에서 제안한 기법의 효율성을 확인하기 위하여 사람이 출입문을 열고 들어오는 상황을 고정된 비디오 카메라로 촬영하였으며, 동적 영상내에 존재하게 될 이동 물체들의 동작 특성을 반영한 지식과 주위 환경에 대한 사전 지식을 충분히 활용하여 이들에 대한 검출 및 추적 작업을 수행하였다. 한편, 본 논문의 실험에서는 동적 영상내에 나타날 수 있는 이동 물체들은 사람과 출입문으로 한정하였으며 이들에 관련된 처리는 각기 독립적으로 행해졌다. 그림 14는 실험에 사용된 연속적인 시간의 흐름에 따른 동적 장면의 변화를 나타내는 영상들이다.

그림 9는 출입문의 추적 개시를 위한 참조 프레임의 예로서 출입문이 닫혀있는 주위 환경에 대한 정보를 그림 2와 같은 형태로 기술하도록 고안되었다. 이 참조 프레임은 수직선(vertical line)에 의하여 문과 벽을 정의하고 있다.

```

Frame DOOR_REFERENCE With
VERTICAL_LINE OF WALL_REGION
  position      : (Xmin,Ymin) < (x,y) < (Xmax,Ymax)
  length        : Lmin < l < Lmax
  Mean intensity : Mmin < m < Mmax
  Contrast of intensity : Cmin < c < Cmax
VERTICAL_LINE OF DOOR_REGION
  position      : (Xmin,Ymin) < (x,y) < (Xmax,Ymax)
  length        : Lmin < l < Lmax
  Mean intensity : Mmin < m < Mmax
  Contrast of intensity : Cmin < c < Cmax
end frame
    
```

그림 9. 출입문에 대한 참조 프레임의 예  
Fig. 9. Example of reference frame for door.

즉, 벽을 대표하는 특성은 하나의 수직선으로서, 그 수직선은  $(X_{min}, Y_{min})$  과  $(X_{max}, Y_{max})$  사이에 위치하여야 하며, 길이는  $L_{min}$  과  $L_{max}$  에 의하여 그리고 평균 밝기값은  $M_{min}$  과  $M_{max}$  에 의하여 제한되고, 수직선에 의하여 분리되는 좌 우 영역의 밝기값의 대조는  $C_{min}$  과  $C_{max}$  사이의 값을 가져야 한다. 여기서 평균 밝기값은 수직선 주위의 화소값으로부터 구해지며 밝기값의 대조는 수직선을 중심으로한 출입문과 벽 영역의 밝기차를 통해 구해진다. 유사한 방법으로 문에 대한 사전 지식은 수직선의 위치, 길이, 평균 밝기값 및 밝기값의 대조를 나타내는 슬롯과 슬롯에 할당될 수 있는 슬롯값의 범위에 의하여 정의된다. 이상과 같은 조건을 만족하는 특성값들이 입력 영상에서 확인되면 이들에 의하여 참조 프레임의 슬롯값들은 채워지게 되고, 이때 모든 슬롯값들이 채워진 참조 프레임을 예시화된 참조 프레임(instantiated reference frame)이라 부른다.

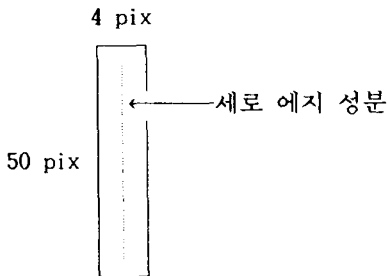


그림 10. 수직 성분 검출을 위한 탐색 마스크  
Fig. 10. Search mask for detecting vertical lines.

수직 성분을 검출하기 위하여 참조 프레임의 position 슬롯에 의하여 제한된 영역에 대하여 수직 방향으로 에지 연산자를 적용한 후, 그림 10과 같은 탐색 마스크(4×50)를 제한된 영역안을 이동하면서 추출된 수직 에지의 갯수를 측정한다. 즉, 추출된 수직 에지의 갯수가 적절한 임계값(예, 45) 이상이 되는 탐색 마스크의 위치를 수직 성분의 위치로 간주한다.

활성화된 참조 프레임의 각 슬롯에 할당된 값들이 부적합하다고 판정되면, 즉 참조 프레임의 예시화가 성공적이지 못하면, 출입문의 열림에 의한 장면의 변화가 생기고 있음을 가정하고 출입문에 대한 검출과 추적 작업을 수행한다. 출입문의 이동을 검출 및 추적하기 위한 프레임 시스템의 기본 구조는 그림 5와 같은 형태로 설계되었으며, 이동 물체에 대한 물체 프레임의 기본 구조는 그림 3 그리고 물체의 이동 상태를 나타내는 동작 프레임의 기본 구조는 그림 4와 같은 형태로 설계되었다. 그림 11은 이동 물체가 출입문인 경우에 물체 프레임의 예를 보여준다.

```

Frame DOOR with
  Part_of      : [name of action frames]
  Preprocessor_1 : [EDGE_DETECTOR()]
  VERTICAL_LINE
    length      : [number of edges] > Lmin
    Edge_direction : 1 < [direction_of_edges] < 3
    Approximate_position : [average position of vertical edges]
end frame
    
```

그림 11. 출입문에 대한 물체 프레임의 예  
Fig. 11. Structure of door frame.

그림 11에서 Part\_of 슬롯은 상위 레벨의 동작 프레임과의 연결을 나타내는 슬롯으로서, 슬롯값은 동작 프레임의 이름(즉, OPEN, CLOSE, STOP)이 된다. Preprocessor\_1 슬롯에는 절차적 지식(procedural knowledge)으로서 에지 검출기가(edge detector)명시되어 있다. 따라서 출입문에 대한 물체 가설은 활성화된 프레임의 Preprocessor\_1 슬롯에 있는 에지 검출기를 상위 레벨의 동작 프레임에서 전달받은 CLUE\_SEARCH\_REGION에 적용함으로써 검증된다. 그림 10의 예에서는 출입문을 수직선(VERTICAL\_LINE)에 근거하여 정의하고 있다. 즉, 길이가  $L_{min}$  보다 크고 에지들의 방향값(edge direction)이 1에서 3 사이의 값을 갖는 수직선의 위치 정보가 Approximate\_position 슬롯에 등록된다.

본 논문에서는 앞에서 언급한 바와 같이 동작 가설을 생성하기 위하여 지식 베이스의 가설 생성 규칙을 사용한다. 참조 프레임의 예시화가 실패하게 되면,

바로 출입문의 동작 가설을 생성하는 작업이 다음과 같은 규칙들에 의하여 구동된다.

- Rule-1. IF (previous activated frame = DOOR\_REFERENCE) && (its instantiation = unsuccessful) THEN (activate OPEN frame)
- Rule-2. IF (previous activated frame = OPEN) && (its instantiation = successful) THEN (activate OPEN frame)
- Rule-3. IF (previous activated frame = OPEN) && (its instantiation = unsuccessful) THEN (activate CLOSE frame)
- Rule-4. IF (previous activated frame = CLOSE) && (its instantiation = successful) THEN (activate CLOSE frame)
- Rule-5. IF (previous activated frame = CLOSE) && (its instantiation = unsuccessful) THEN (activate STOP frame)
- Rule-6. IF (previous activated frame = STOP) && (its instantiation = successful) THEN (activate STOP frame)
- Rule-7. IF (previous activated frame = STOP) && (its instantiation = unsuccessful) THEN (activate OPEN frame)

각 규칙의 전제부는 활성화되었던 프레임의 이름을 확인하는 함수(function)와 그 프레임에 대한 예시화의 결과를 확인하는 함수로 구성되고, 결론부는 다음에 활성화시킬 프레임의 이름을 확인하는 함수로 구성된다. 위에 정의된 규칙들은 '물체의 동작 상태는 급격한 변화를 하지 않는다'는 전제하에 작성되었다. 따라서, 예시화에 성공한 프레임이 연속적으로 활성화된다. Rule-3, Rule-5, 그리고 Rule-7은 활성화된 프레임의 예시화가 실패할 경우, 다음에 활성화시킬 프레임을 나타낸다. 즉, OPEN 프레임의 예시화가 실패한 경우에는 우선적으로 CLOSE 프레임의 활성화를 시도하고, 활성화된 CLOSE 프레임의 예시화도 실패하는 경우에는 STOP 프레임의 활성화를 시도하게 된다.

본 논문에서는 출입문의 추적 작업을 출입문의 우측 상단부에 위치하는 수직 성분에 의하여 수행하였다. 그림 15는 그림 13의 입력 영상들에 대한 출입문의 이동 상황을 추적한 결과를 보여준다. 즉, 동작 가설에 의하여 정의된 CLUE\_SEARCH\_REGION 내에서 에지 연산자를 적용한 결과, 그리고 탐색 마스크를 적용하여 검출된 수직 성분을 보여준다. 탐색 마스크의 적용 범위는 CLUE\_SEARCH\_REGION의 상단부로 한정하였다. 이는 사람의 출입에 의하여 수직 성분의 에지들이 CLUE\_SEARCH\_REGION의 하단부에 생성될 수 있기 때문이다. 수직 성분의 검출 작업은 탐색 마스크를 CLUE\_SEARCH\_REGION의 우측에서 좌측으로 이동하면서 수행되며, 일단 수직 성분의 존재가 확인되면 탐색 마스크의 이동 작업을 멈추도록 하였다. 이는 출입문 내부에 존재하는 또 다른 수직 성분들이 검출되는 것을 방지하기 위함이다.

출입문을 열고 들어오는 사람의 움직임을 검출 및 추적하는 작업은 출입문이 닫혀진 상태에서 열리기 시작하고 사람이 나타나게 될 때, 그 존재는 출입문에 인접한 영역에 위치한다는 전제를 이용한다. 즉, 초기 사람의 존재 영역을 출입문 주위로 제한하고, 출입문이 열리면서 만들어지는 복도 영역과 만나는 출입문 부근의 명암 분포를 조사함으로써 사람의 출현 여부를 판단한다. 출입문의 경우와 마찬가지로 본 논문에서 사람에 대한 추적 작업의 개시를 위해 사용되는 참조 프레임은 그림 12와 같다.

```

Frame MAN_REFERENCE With
VERTICAL LINE OF WALL_REGION
  position      : (Xmin,Ymin) < (x,y) < (Xmax,Ymax)
  length       : Lmin < l < Lmax
  Contrast of intensity : Cmin < c < Cmax
VERTICAL LINE OF DOOR_REGION
  position      : (Xmin,Ymin) < (x,y) < (Xmax,Ymax)
  length       : Lmin < l < Lmax
  Contrast of intensity : Cmin < c < Cmax

Difference of position : d = |position - dposition|
Homogeneous intensity : h < Hmax
and frame
    
```

그림 12. 사람에 대한 참조 프레임의 예  
Fig. 12. Example of reference frame for man.

그림 12의 참조 프레임(MAN\_REFERENCE)에 포함된 슬롯의 형태는 그림 9의 출입문을 위한 참조 프레임(DOOR\_REFERENCE)과 유사하나 사용된 참조 영상이 출입문이 닫혀진 상태가 아닌 열려진 상태를 갖는다는 점에서 차이가 있다. 따라서 그림 12의 두 Contrast of intensity 슬롯은 각각 벽과 복도 영역 그리고 출입문과 복도 영역간의 명암 대조를

의미한다. 그리고 *difference of position*은 수직선으로 정의된 벽과 출입문 간의 위치 차를 나타내는 슬롯으로서, 결국 참조 영상에서의 복도 영역에 해당되며 *Homogeneous intensity* 슬롯은 바로 이 복도 영역에 대한 명암 분포의 균일 정도를 의미한다. 따라서 출입문이 열리고 사람이 나타날 경우 복도 영역의 명암 분포는 고르지 못하게 되고 *Homogeneous intensity* 슬롯을 제한된 범위의 값으로 채우지 못하게 된다. 결국 이 참조 프레임은 예시화에 성공하지 못하게 되고 이는 출입문의 경우와 마찬가지로 사람의 검출 및 추적 작업의 개시를 뜻하게 된다.

본 논문에서 사람의 움직임은 나타내는 동작 개념은 감시 카메라를 기준으로하는 이동 방향에 따라 RIGHTWARD, LEFTWARD, FORWARD, BACKWARD 등과 같은 4 방향에 근거한 동작 방향 프레임과 STANDING 프레임으로 정의하였으며 이들 프레임들은 그림 5의 프레임간 계층 구조에서 상위 레벨에 해당한다. 각 동작 프레임의 기본 구조는 그림 4와 동일하며, 현 영상에서의 이동 물체의 위치와 전 영상에서의 이동 물체의 위치를 비교하여 동작 상태의 적합성을 검증하는 *Relation* 슬롯에서는 전 영상 및 현 영상에서 추출된 사람에 해당되는 영역의 위치 관계(*spatial relation*)를 확인한다. 예를 들어, RIGHTWARD 프레임의 <Relation> 슬롯은 다음과 같이 정의된다.

```
Relation : RIGHT
          (MER(man_in_current_image),
           MER(man_in_previous_image))
```

```
Frame MAN with
.
Part_of      : [name of motion frames]
Preprocessor_2 : [BINARY_SEGMENTOR()]
Preprocessor_3 : [REGION_ANALYZER()]

Mean_intensity : [integer value] such that (? < Mmax)
Contrast_value : [integer value] such that (? > Cmin)
Area          : [number of pixels] such that (? > Amin)

Lower_position of MER : [2D coordinates]
Upper_position of MER : [2D coordinates]
Centroid of MER      : [2D coordinates]
Size of MER          : [size of number of pixels]

Area_ratio          : [Area + Size = real value]
                    such that (? > ARmin)
end frame
```

그림 13. 사람에 대한 물체 프레임의 구조

Fig. 13. Structure of man frame.

즉, 현 영상에서 추출된 사람에 해당되는 영역에 대한 최소 포함 사각형(MER: Minimum Enclos-

ing Rectangle)의 위치는 전 영상에서 추출된 사람에 대응되는 영역에 대한 최소 포함 사각형보다 오른쪽에 위치하여야 함을 나타낸다. 이동 물체인 사람(MAN)에 대한 물체 프레임의 예는 그림 13과 같이 설계하였다.

그림 13의 프레임은 사람에 대한 예를 입력 영상에서 확인하기 위하여 <Preprocessor\_2> 와 <Preprocessor\_3> 을 사용할 것을 지시한다. Preprocessor\_2는 상위 레벨인 동작 프레임의 CLUE SEARCH REGION에 의하여 제한된 영역에서 히스토그램을 작성한 후, 작성된 히스토그램을 분석하여 이진 영상(binary image)을 형성하여 레이블링(labeling)을 수행하는 연산자이다. 이진 영상을 형성하는 과정은 제 3절에서 기술한 방법을 사용한다. <Preprocessor\_3>은 <Preprocessor\_2>에 의하여 추출된 영역을 분석하여 각 슬롯에 대응되는 특성을 추출하는 연산자이다. 예를 들어, 추출된 영역의 평균 밝기값(mean\_intensity), 주위 영역과의 밝기값의 대조(contrast\_value), 면적(area), 그리고 추출된 영역을 포함하는 최소 포함 사각형(MER) 등과 같은 특성값들이 Preprocessor\_3에 의하여 추출된다.

출입문의 경우와 마찬가지로 사람의 이동에 대한 동작 가설을 생성하기 위하여 지식베이스의 가설 생성 규칙을 사용한다. 사람에 관한 참조 프레임이 예시화에 실패하게 되면 사람의 동작 가설을 생성하는 작업이 다음의 규칙들에 의하여 구동된다.

- Rule-1. IF (previous activated frame = MAN\_REFERENCE) && (its instantiation = unsuccessful) THEN (activate STANDING frame)
- Rule-2. IF (previous activated frame = STANDING) && (its instantiation = successful) THEN (activate STANDING frame)
- Rule-3. IF (previous activated frame = STANDING) && (its instantiation = unsuccessful) THEN (activate RIGHTWARD frame)
- Rule-4. IF (previous activated frame = RIGHTWARD) && (its instantiation = successful) THEN (activate RIGHTWARD frame)
- Rule-5. IF (previous activated frame = RIGHTWARD) && (its instantiation = unsuccessful)



- THEN (activate FORWARD frame)
- Rule-6. IF (previous activated frame = FORWARD) &&  
(its instantiation = successful)  
THEN (activate FORWARD frame)
- Rule-7. IF (previous activated frame = FORWARD) &&  
(its instantiation = unsuccessful)  
THEN (activate LEFTWARD frame)
- Rule-8. IF (previous activated frame = LEFTWARD) &&  
(its instantiation = successful)  
THEN (activate LEFTWARD frame)
- Rule-9. IF (previous activated frame = LEFTWARD) &&  
(its instantiation = unsuccessful)  
THEN (activate BACKWARD frame)
- Rule-10. IF (previous activated frame = BACKWARD) &&  
(its instantiation = successful)  
THEN (activate BACKWARD frame)
- Rule-11. IF (previous activated frame = BACKWARD) &&  
(its instantiation = unsuccessful)  
THEN (activate STANDING frame)

각 규칙의 전제부 및 결론부의 구조는 출입문의 경우와 동일하다. 위에 정의된 규칙들 역시 물체의 동작 상태는 급격한 변화를 하지 않는다는 전제하에 작성되었다. 따라서, 예시화에 성공한 프레임이 연속적으로 활성화된다. Rule-3, Rule-5, Rule-7, Rule-9 그리고 Rule-11은 활성화된 프레임의 예시화가 실패한 경우에 다음에 활성화시킬 프레임을 나타낸다. 즉, STANDING 프레임의 예시화가 실패한 경우에는 우선적으로 RIGHTWARD 프레임의 활성화를 시도하고, 활성화된 RIGHTWARD 프레임의 예시화도 실패하는 경우에는 FORWARD 프레임의 활성화를 시도하게 된다. FORWARD 프레임 역시 예시화에 실패하는 경우에는, LEFTWARD 프레임을 활성화시키고 다시 이 프레임도 예시화에 실패할 경우 최종적으로 BACKWARD 프레임의 예시화를 시도하게 된다.

그림 16과 그림 17은 그림 13의 입력 영상들에서 출입문을 열고 들어오는 사람의 이동 상황을 추적한 결과를 보여준다. 그림 16은 동작 가설 생성시 CLUE\_SEARCH\_REGION에 의하여 제한된 탐색

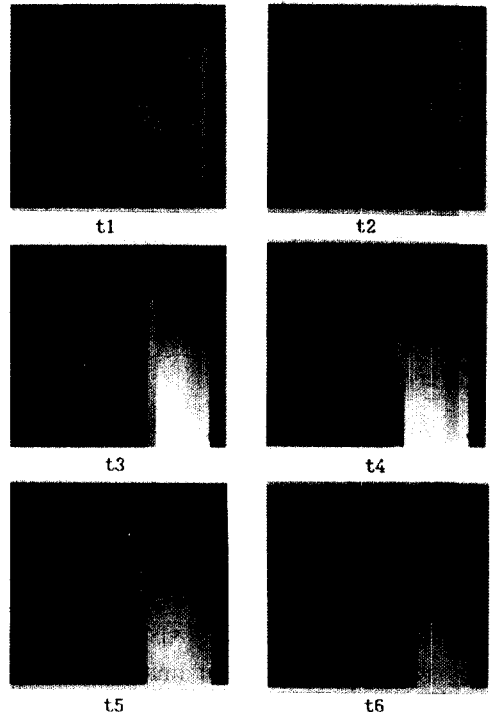
영역을 보여주고, 그림 17은 제한된 탐색 영역안에서 사람에 대한 물체 프레임을 예시화하여 검출된 최소 포함 사각형을 보여준다.

### V. 결론(Conclusion)

본 논문에서는 동적 영상내 이동 물체의 검출과 추적을 위하여 문제 영역에 대한 사전 지식과 물체 및 동작 지식을 활용하는 방법을 제안하였다. 즉, 예상되는 물체 가설(object hypotheses)과 동작 가설(motion hypotheses)을 생성하고 이를 검증함으로써 제한된 탐색 영역내에서 효율적으로 이동 물체를 검출하고 추적하는 방법을 제안하였다.

본 논문에서 채택한 가설 생성 및 검증 방법은 완전한 하향식 제어 방식이다. 즉, 하나의 동작 가설에 대한 검증이 완료된 후 비로서 다른 동작 가설을 생성하게 되고, 물체 가설은 동작 가설에 의존하여 생성된다. 이러한 제어 방식은 시스템의 융통성을 위축시키는 단점은 있으나, 처리 과정이 단순하여 시스템의 효율성을 높이는 효과를 갖는다. 특히, 물체 및 동작 프레임에 대한 정의가 주어진 문제에 적합하게 이루어진 경우에는 매우 신속히 이동 물체에 대한 검출 및 추적 작업이 수행될 수 있다.

본 논문에서는 영상내에 존재하는 복수개의 이동 물체에 대한 처리를 동시에 수행하지 않고 독립적으



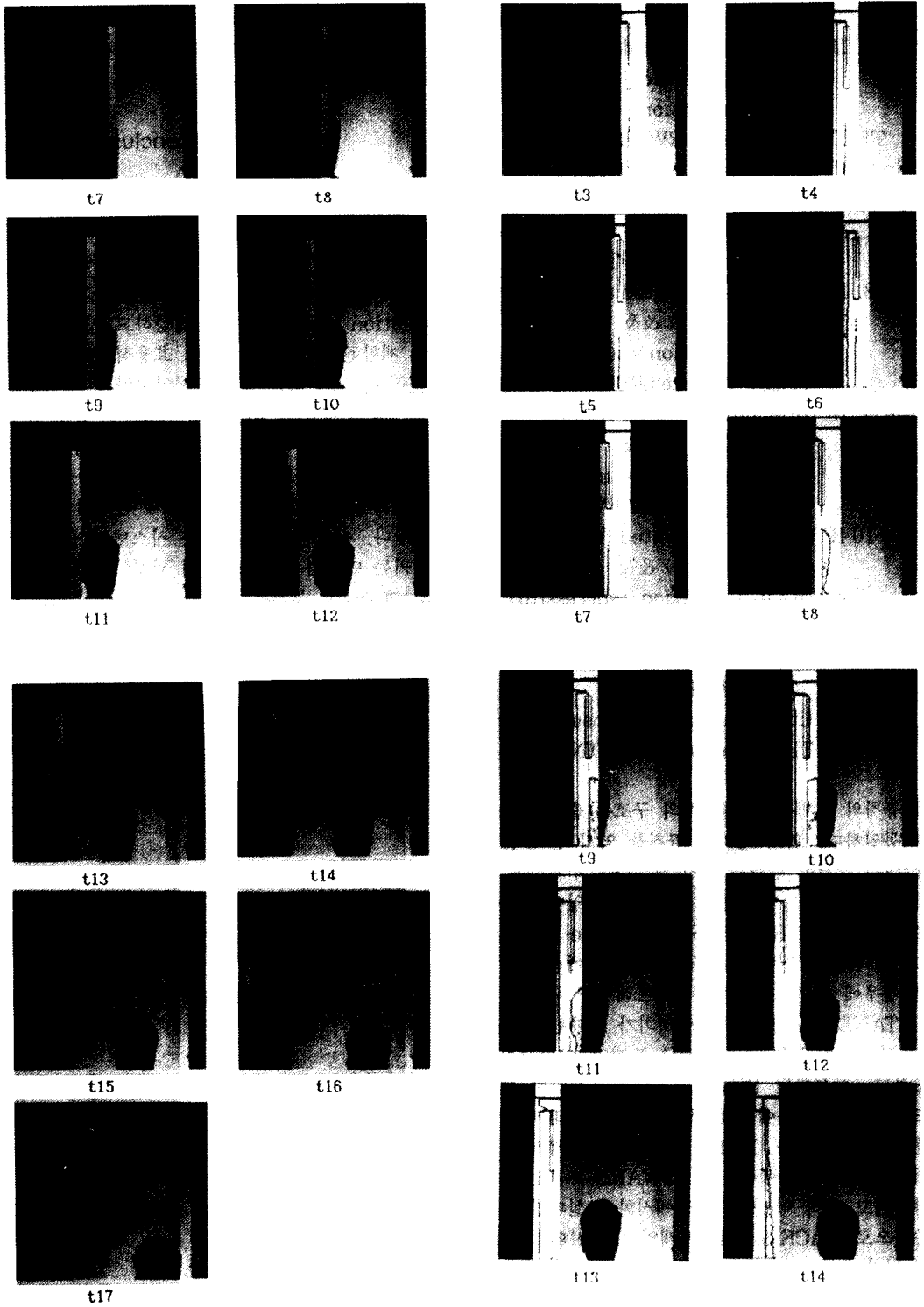


그림 14. 시간의 변화에 따른 입력 영상들의 예  
 fig. 14. Successive input images along time sequence.

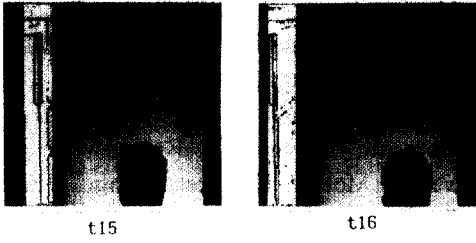


그림 15. 탐색 마스크에 의한 출입문의 추적  
Fig. 15. Tracking moving dorr by search mask.

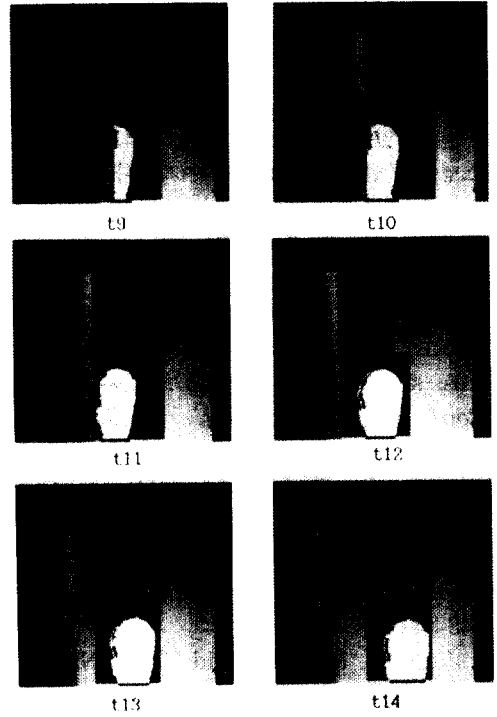
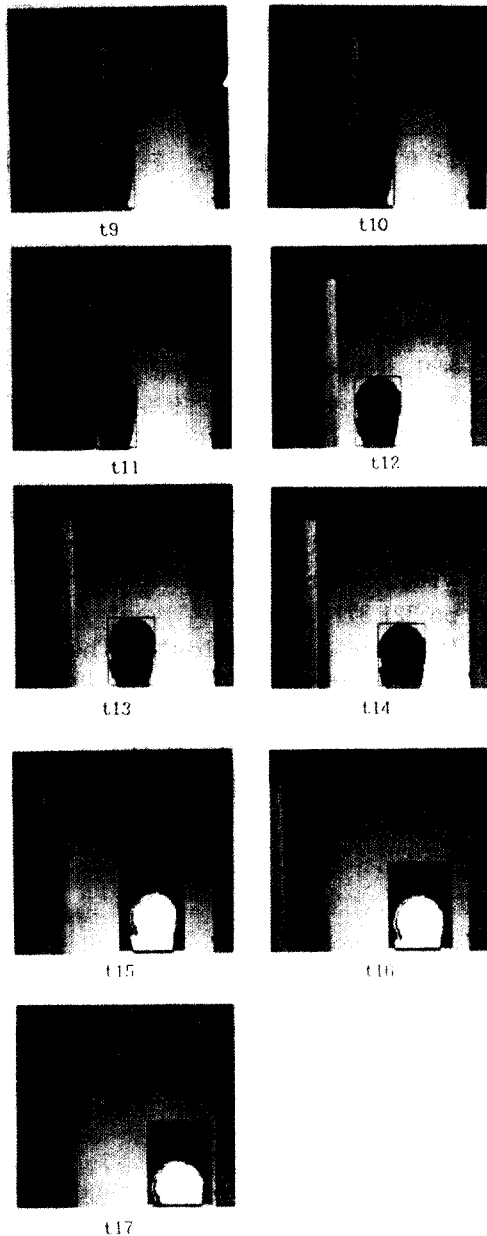


그림 16. 동작 가설에 의하여 제한된 탐색 영역  
Fig. 16. CLUE\_SEARCH\_REGIONs determined by motion hypotheses.

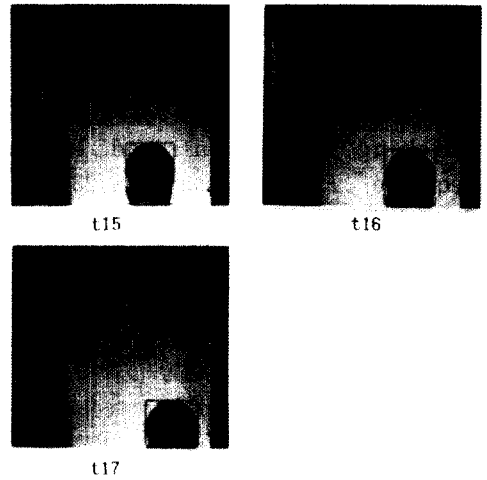


그림 17. 최소 포함 사각형에 의한 사람의 추적  
Fig. 17. Tracking man by minimum enclosing rectangle.

로 수행하였다. 이러한 문제점은 동시 프로그래밍 (concurrent programming) 기법을 활용하고 가설 생성 규칙의 구조를 개선할 경우 즉, 규칙의 전체부에서 이전 가설의 검증에 대한 다양한 분석을 시도하면, 보완될 수 있으며 복잡한 상황에서도 적응성이 높은 시스템이 구현될 수 있으리라 여겨진다.

#### 參 考 文 獻

- [1] J. L. Potter, "Scene segmentation by velocity measurements obtained with a cross-shaped template," *Proc. 4th Int. Conf. Artificial Intell.*, pp.803 810, 1975.
- [2] William B. Thompson, "Combining Motion And Contrast For Segmentation," *IEEE Trans. Pattern Anal. And Machine Intel.*, vol.PAMI-2, No.6, pp. 543 549, NOV. 1980.
- [3] R. Jain, D. Militzer, and H. H. Nagel, "Separating non-stationary from stationary scene components in a sequence of real world TV-images," *Proc. 5th Int. Joint Conf. Artificial Intell.*, pp.612 618, 1977.
- [4] W. K. Chow and J. K. Aggarwal, "Computer analysis of planar curvilinear moving images," *IEEE Trans. Comput.*, vol.C-26, pp.179 185, 1977.
- [5] Hsi Jian, Lung Fa Huang, Zen Chen, "Multi-Frame Ship Detection And Tracking In An Infrared Image Sequence," *Pattern Recog.*, vol.23, No. 7, pp.785 798, 1990.
- [6] Masahiko Yachida, Minoru Asada, Saburo Tsuji, "Automatic Analysis of Moving Images," *IEEE Trans. Pattern Anal. And Machine Intel.*, vol.PAMI-3, No.1, pp.12 20, JAN. 1981.
- [7] John K. Tsotsos, John Mylopoulos, H. Dominic Covvey, "A Framework for Visual Motion Understanding," *IEEE Trans. Pattern Anal. And Machine Intel.*, vol.PAMI-2, No.6, pp.563 573, NOV. 1980.
- [8] Frank P. Ferrie, Martin D. Levine, "Cell Tracking: A Modeling and Minimization Approach," *IEEE Trans. Pattern Anal. And Machine Intel.*, vol.PAMI-4, No.3, pp.277 291, MAY 1982.
- [9] 최형일, 컴퓨터 비전 입문, 홍릉과학 출판사, pp.171 202, 1991.
- [10] C. Andrew, Robotics and AI, Prentice-hall Inc., pp.80 90, 1987.
- [11] 방수남, 최형일, 유석인, "지식에 기반을 둔 컴퓨터 시각 시스템에 있어서 지식 표현 및 제어 전략에 관한 연구: 흑판 모델을 이용한 항공 사진 해석 시스템," *정보과학회 논문지*, Vol.17, No.5, pp.605 618, SEP. 1990.
- [12] Hwang V.S. Davis and T. Matshyama, "Hypothesis integration in image understanding System," *Computer Vision Graphics and Processing*, vol.36, pp.321 371, 1986.
- [13] D. Marr & E. Hildreth, "Theory of edge detection," *Proc. of Royal Society of London*, vol.207, pp.187 217, 1980.
- [14] D.H. Ballard & C.M. Brown, *Computer Vision*, Prentice Hall, Chap. 7, 1982.

著者紹介



李眞鎬(準會員)

1967年 2月 3日生. 숭실대학교 공과대학 전자계산학과 졸업(공학사). 1993年 2月 숭실대학교 대학원 전자계산학과 졸업(공학석사). 주관심분야는 컴퓨터 비전, 퍼지 이론, 지식기반 시스템 등임.



崔潑一(正會員)

1956年 8月 16日生. 1979年 2月 연세대학교 전자공학과 졸업(공학사). 1982年 6月 미시간대학교 전산공학과 졸업(공학석사). 1987年 6月 미시간대학교 전산공학과 졸업(공학박사). 1987年 ~ 현재 숭실대학교 전산과 부교수 재직중. 주관심분야는 컴퓨터 비전, 퍼지 및 신경망이론, 패턴인식, 지식기반시스템 등임.