

동적 프로그래밍을 이용한 특징점 정합

김 동 근[†]

요 약

본 논문에서는 기준영상과 탐색영상 사이의 대응되는 특징점을 정합하는 알고리즘을 제안한다. 두 영상에서 특징점을 찾기 위하여 Harris의 코너점 검출기를 사용하였다. 기준영상의 각 특징점에 대해, 정규상관계수가 임계치 이상인 탐색영상의 특징점들로 후보 정합점을 구한다. 최종적으로 동적 프로그래밍을 사용하여 후보 정합점들 중에서 대응되는 특징점을 구한다. 실험으로 인위적인 영상과 실제 영상에서 특징점을 정합하는 결과를 보였다.

Matching Of Feature Points using Dynamic Programming

Dong Keun Kim[†]

ABSTRACT

In this paper we propose an algorithm which matches the corresponding feature points between the reference image and the search image. We use Harris's corner detector to find the feature points in both image. For each feature point in the reference image, we can extract the candidate matching points as feature points in the search image which the normalized correlation coefficient goes greater than a threshold. Finally we determine a corresponding feature points among candidate points by using dynamic programming. In experiments we show results that match feature points in synthetic image and real image.

키워드 : 특징점(feature points), 정규상관계수(normalized correlation coefficient), Harris의 코너점 검출기(Harris's corner detector), 동적 프로그래밍(dynamic programming)

1. 서 론

동일한 장면에 대한 서로 다른 두 영상을 정합하는 문제는 영상등록(image registration), 스테레오 정합(stereo matching), 움직임 추정(motion estimation) 등의 컴퓨터 비전의 여러 분야에서 매우 중요하고 어려운 문제 중에 하나이다.

영상정합 방법은 크게 템플릿 정합(template matching)과 특징정합(feature matching)의 두 종류로 분류할 수 있다 [1]. 템플릿 정합 방법은 윈도우 템플릿을 기반으로 정규상관계수가 높은 곳을 정합 점으로 결정하거나, 영상 밝기 차이의 절대값 합이 작은 곳을 정합 점으로 결정하는 방법이다. 특징정합 방법은 영상에서 에지, 코너, 라인 등과 같은 뚜렷한 특징을 먼저 검출하고, 검출된 특징들의 기하학적 관계를 정의하여 트리 탐색, 이완(relaxation) 기법 등을 통하여 특징을 정합하는 방법이다.

카메라 캘리브레이션된 스테레오 영상정합에서는 에피폴라 제약조건(epipolar constraints)을 정합에 이용한다[1, 2, 4]. 캘리브레이션되지 않은 스테레오 영상정합에 에피폴라

제약조건을 계산하기 위하여 정합되는 대응점들을 사용하여 기본행렬(fundamental matrix, F)를 정확히 계산하는 것이 중요한 문제이다. 시퀀스 영상에서는 인접한 프레임 사이에 시간간격이 작고, 물체의 움직임이 급격하지 않다는 가정을 하여 이전 프레임 위치의 근처에서 대응되는 정합점을 탐색한다.

본 논문에서는 정합될 기준영상(reference image)과 탐색영상(search image)에서 특징점을 추출하고 특징점들 사이의 정합되는 대응점(correspondence points)을 동적 프로그래밍(dynamic programming)을 사용하여 효율적으로 결정하는 방법을 제안하였다. 특징점 추출 및 후보점(candidate points) 추출 방법은 Z.ZHANG[1, 2]과 유사하게 Harris에 의해 제안된 Plessey 코너점 추출 방법을 사용하였으며, 후보점 추출을 위해 정규상관계수를 사용하였다. 본 논문에서는 반복적인 이완기법을 사용하지 않고 기준영상의 특징점의 그래프와 유사한 탐색영상의 후보점에 의한 그래프를 동적 프로그래밍을 이용하여 찾는 방법으로 정합되는 대응점을 효율적으로 구하였다.

본 논문의 구성은 다음과 같다. 2장에서 기존의 특징점 정합 방법을 설명하고, 3장에서 본 논문에서 제안한 동적

[†] 종신회원 : 천안공업대학 컴퓨터과 교수
논문접수 : 2002년 8월 2일, 심사완료 : 2002년 11월 9일

프로그래밍에 의한 특징점 정합 방법을 설명하며, 4장에서는 실험 결과를 보이고, 5장에서 결론 및 향후 연구에 대해 기술한다.

2. 관련 연구

2.1 특징점 추출

영상의 특징점(feature points)은 에지 또는 코너점 등과 같이 주변 화소에 비해 뚜렷이 구분되는 점으로, 영상표현, 영상정합, 스테레오 비전, 물체추적 등의 많은 분야에서 중요한 역할을 한다. 대표적인 코너점 추출 방법으로 주변화소에 대한 밝기 차를 이용하는 SUSAN 방법[3]과 영상의 그래디언트 값을 이용하는 Plessey 코너점 추출 방법[1, 8]이 있다. Harris에 의해 제안된 Plessey 코너점 추출 방법은 윈도우 영역에서 영상의 그래디언트 값에 의한 공분산 행렬(covariance matrix) C 를 사용한다. 영상을 가우시안 필터링하여 \hat{I} 를 구하고, x, y 방향의 편미분 \hat{I}_x, \hat{I}_y 를 계산한다. 화소 (x, y) 를 중심으로 제한된 윈도우 내에서 편미분의 합을 이용하여 공분산 행렬 C 를 구한 다음, 가우시안 커널을 이용하여 윈도우의 중앙에 위치한 화소들의 미분값에 더 큰 가중치를 주어 합을 구한다.

$$C = \begin{bmatrix} \langle \hat{I}_x^2 \rangle & \langle \hat{I}_x \hat{I}_y \rangle \\ \langle \hat{I}_x \hat{I}_y \rangle & \langle \hat{I}_y^2 \rangle \end{bmatrix} \quad (1)$$

공분산 행렬 C 의 두 고유치(eigenvalue) λ_1, λ_2 가 모두 큰 값일 때 코너점으로 판단한다. Tomasi와 Kanade[11]는 두 고유치중 작은 값이 임계치보다 크면 코너점으로 판단하였다. Harris는 코너점 정도(corner strength)를 다음과 같이 정의하였다.

$$\begin{aligned} R_1(x, y) &= \text{trace}(C) / \det(C) \\ &= (\langle \hat{I}_x^2 \rangle + \langle \hat{I}_y^2 \rangle) / (\langle \hat{I}_x^2 \rangle \langle \hat{I}_y^2 \rangle - \langle \hat{I}_x \hat{I}_y \rangle^2) \\ &= (\lambda_1 + \lambda_2) / \lambda_1 \lambda_2 \end{aligned} \quad (2)$$

코너점 정도 $R_1(x, y)$ 가 임계치 보다 작은 화소 (x, y) 를 코너점으로 추출한다. Harris의 개선된 Plessey 코너점 검출 방법은 코너점의 정도를 다음과 같이 계산하였다.

$$\begin{aligned} R_2(x, y) &= \det(C) - k \text{trace}^2(C) \\ &= \langle \hat{I}_x^2 \rangle \langle \hat{I}_y^2 \rangle - \langle \hat{I}_x \hat{I}_y \rangle^2 - k(\langle \hat{I}_x^2 \rangle + \langle \hat{I}_y^2 \rangle)^2 \end{aligned} \quad (3)$$

k 는 강한 스텝 에지를 억제하기 위한 상수 값으로 Z.ZHANG 등은 실험에서 $k=0.04$ 를 사용하였다. 각 화소 (x, y) 의 코너점 정도(corner strength)인 $R_2(x, y)$ 를 공분산 행렬

C 를 이용하여 구하고, $R_2(x, y)$ 가 임계치 보다 큰 화소 (x, y) 를 코너점으로 추출한다.

2.2 특징점 정합

기준영상에서 추출된 특징점에 대응하는 탐색영상의 특징점을 구하는 문제는 서로 다른 시야에서 획득한 스테레오 영상으로부터 카메라 정보 계산(camera calibration), 깊이 정보(depth information) 계산, 3D 구조를 복구(reconstruction of 3D structure) 등의 스테레오 비전 및 시퀀스 영상으로부터 움직임 추정(motion estimation), 물체추적(object tracking), 영상 등록(image registration) 등의 영상처리 및 컴퓨터 비전의 많은 문제에서 매우 중요한 문제이다.

기준영상의 특징점에 대한 탐색영상의 특징점 중에 대응되는 점을 찾기 위하여 주로 사용하는 방법이 고정된 크기의 윈도우에 의한 정규상관계수(normalized correlation coefficient)를 이용하여, 정규상관계수 값이 가장 큰 탐색영상의 특징점을 정합되는 대응점으로 정하는 방법이다. $(2W+1) \times (2W+1)$ 의 윈도우 크기에서 기준영상의 특징점 $p(u_1, v_1)$ 와 탐색영상의 특징점 $q(u_2, v_2)$ 사이의 정규상관계수 NCC 는 다음과 같다.

$$\begin{aligned} NCC(p, q) &= \frac{1}{K} \sum_{i=-W}^W \sum_{j=-W}^W [I_1(u_1+i, v_1+j) - \bar{I}_1(u_1, v_1)] \\ &\quad \times [I_2(u_2+i, v_2+j) - \bar{I}_2(u_2, v_2)] \\ K &= (2W+1)(2W+1) * \sigma(I_1(u_1, v_1)) * \sigma(I_2(u_2, v_2)) \end{aligned} \quad (4)$$

위에서 $\bar{I}_1(u_1, v_1), \bar{I}_2(u_2, v_2)$ 는 윈도우 영역 내에서의 각 영상의 평균이고, $\sigma(I_1(u_1, v_1)), \sigma(I_2(u_2, v_2))$ 는 윈도우 영역 내에서의 표준편차이다.

기준영상의 한 특징점 $p_i \in P$ 에 대한 탐색영상의 대응되는 특징점 $q_j \in Q$ 를 정하는 가장 간단한 방법은 정규상관계수 값이 가장 큰 q_j 를 p_i 의 대응점으로 하는 방법이다.

$$\max_{q_j \in Q} NCC(p_i, q_j) \quad (5)$$

정규상관계수 값이 가장 큰 특징점을 대응점으로 정할 때 윈도우를 크게 하고, 영상의 밝기 값의 특징이 뚜렷할 때 대부분의 경우에 만족할 만한 결과를 얻는다. 그러나 제한된 윈도우를 사용할 때는 정규상관계수 값이 가장 큰 값을 갖는 특징점이 올바른 대응점이 아닐 수 있다. 또한 기준영상의 특징점에 대한 대응되는 탐색영상의 특징점이 없는 경우와 탐색영상에 있는 하나의 특징점에 대해 기준영상의 둘 이상의 특징점이 대응될 수 있는 애매성이 발생할 수 있다. 이러한 모든 문제에 대한 최적해를 구하기는 어렵

다. 식 (5)에 의해 결정된 대응점 q_j 에 대해, 역으로 기준영상의 특징점들 중에 가장 큰 정규상관계수를 갖는 특징점이 p_i 일 때 q_j 를 p_i 의 대응점으로 결정하는 방법을 사용하여 좀더 정확성을 높일 수 있다.

다른 방법은 대응점의 후보점(candidate points)을 먼저 구하고, p_i 와 q_j 가 대응점이면 이들의 이웃에 있는 특징점들 사이에 많은 대응점들이 있을 것이라는 정보를 이용한다[1]. 기준영상의 주어진 특징점에 대한 탐색영상의 대응되는 후보점들을 구하고 이들 사이에 정합강도(strength of the match)을 계산하여, 정합강도를 최대로 하는 후보점들 중의 하나를 대응점으로 결정하는 방법을 반복적 이완(relaxation) 기법으로 구한다.

기준영상의 각 특징점 p_i 에 대해 탐색영상의 특징점들 중에 $NCC(p_i, q_j) > 0.8$ 인 모든 q_j 들을 p_i 의 후보점(candidate points)으로 구한다. 기준영상의 특징점 p_i 와 탐색영상의 후보점 q_j 사이의 매칭강도(strength of the match), S_M 은 다음과 같이 정한다.

$$S_M(p_i, q_j) = c_{ij} \sum_{r_k \in N(p_i)} \max_{s_l \in N(q_j)} \frac{c_{kl} \delta(p_i, q_j; r_k, s_l)}{1 + \text{dist}(p_i, q_j; r_k, s_l)} \quad (6)$$

$N(p_i)$ 와 $N(p_j)$ 는 p_i 와 q_j 의 반경내의 이웃 특징점들이고, C_{ij} 는 p_i 와 q_j 사이의 정규상관계수 값이고, C_{kl} 은 r_k 와 s_l 사이의 정규상관계수 값이다. $\text{dist}(p_i, q_j; r_k, s_l)$ 는 p_i 와 q_j 의 거리와 r_k 와 s_l 의 거리의 평균이다.

$$\text{dist}(p_i, q_j; r_k, s_l) = (\|p_i - q_j\| + \|r_k - s_l\|) / 2 \quad (7)$$

$\delta(p_i, q_j; r_k, s_l)$ 는 p_i 와 q_j 의 이웃에 있는 특징점 r_k 와 s_l 가 후보 정합점일 때의 기여도로 이웃노드와의 상대거리차이 r 값이 매우 큰 값이면 $\delta(p_i, q_j; r_k, s_l) \rightarrow 0$ 이 되어 기여도를 낮게 하고, 매우 작은 값이면 $\delta(p_i, q_j; r_k, s_l) \rightarrow 1$ 되어 기여도를 크게 한다.

$$\delta(p_i, q_j; r_k, s_l) = \begin{cases} e^{-r/\epsilon_r} & \text{if } (r_k, s_l) \text{ is a candidate} \\ & \text{match and } r < \epsilon_r \\ 0 & \text{otherwise} \end{cases} \quad (8)$$

$$\text{where } r = \frac{|\|p_i - r_k\| - \|q_j - s_l\||}{\text{dist}(p_i, q_j; r_k, s_l)}$$

ϵ_r 은 상대 거리 차이에 대한 임계치이다. 위의 매칭강도 S_M 은 이웃에 있는 특징점들 중에서 상대적인 위치가 유사한 후보 정합점의 수가 많을수록 높은 값을 갖는다. 기준영상의 특징점에 대하여 탐색영상의 특징점에서의 여러 후보점들 중에 \max 연산자를 사용하여 가장 가까운 거리 차이

를 갖는 점만을 고려한다. 이웃에 있는 각 후보 정합점에 대한 기여도를 거리에 의해 가중치 되었다. 매칭강도 $S_M(p_i, q_j) \neq S_M(q_j, p_i)$ 인 비대칭이다. 이것은 기준영상의 여러 특징점 $r_k \in N(p_i)$ 가 탐색영상의 하나의 특징점 $s_l \in N(q_j)$ 에 대응할 때 발생한다. 위에서 합을 계산하기 전에 이러한 상황이 발생하는지 조사하여, 가장 큰 값을 주는 특징점 만을 고려하면 해결될 수 있다. 기타 고려될 수 있는 값으로, 영상면의 회전 각이 θ 보다 작다면 $\overline{p_i r_k}$ 와 $\overline{q_j s_l}$ 사이의 각이 θ 보다 작다는 가정을 사용할 수 있다. 이들의 구현에서는 $R = \text{image_width}/8, c_{ij} = 1, \epsilon_r = 0.3, \theta = 90^\circ$ 를 사용하였다.

에너지 함수, E 를 모든 후보 정합들의 매칭강도 S_M 의 합으로 정의하고, 에너지 함수 E 를 최대로 하는 정합을 이완(relaxation) 기법을 통하여 반복적으로 구한다.

$$E = \sum_{(p_i, q_j)} S_M(p_i, q_j) \quad (9)$$

각 후보 정합점에 대한 매칭강도 S_M 을 식 (6)에 의해 구하고, 전체 에너지를 최대로 하는 정합을 갱신하는 방법을 사용한다. 갱신 방법으로는 각 반복에서 기준영상의 특징점과 탐색영상의 후보점 쌍들 중에 이들이 속한 매칭들 중에 가장 높은 매칭강도를 갖는 후보점들 중에 임의 하나를 올바른 대응점으로 선택한 후에, 이들 두 특징점 들을 모든 후보점에서 제거하는 방법으로 최대화하는 방법 등이 있다. 이 방법은 매우 빨리 수렴하는 반면 지역 최대값을 구할 수 있다.

3. 동적 프로그래밍에 의한 대응점 정합

본 논문은 기준영상과 탐색영상의 특징점을 Harris에 의해 제안된 코너점 추출방법으로 추출하고, 특징점들 사이의 대응점 후보(candidates)를 정규상관계수에 의해 구하였다. 기준 영상에서의 특징점들을 연결된 그래프로 생각하여, 탐색영상의 후보점들 중에서 기준영상의 특징점 들의 연결된 그래프와 유사한 그래프를 동적 프로그래밍 방법을 적용하여 결정한다.

3.1 특징점 추출

본 논문에서는 기준영상과 탐색영상의 특징점을 추출 할 때 에지를 먼저 검출하고, 에지에서만 식 (1)과 식 (3)을 적용하여 코너점 정도 $R_2(x, y)$ 를 계산하였다. 코너점 정도가 주어진 임계치 보다 큰 점들을 먼저 검출한다. 이렇게 검출된 특징점들은 인접한 곳에서 많이 검출되기 때문에 이들을 제거하기 위하여 E. Trucco와 A. Verri[4]와 Tomasi와

Kanade[9]에 의해 사용된 방법과 유사하게 검출된 코너점들 $R_2(x, y)$ 값이 큰 순서로 리스트로 가지고 있으면서, 가장 큰 코너점 정도를 갖는 점부터 조사하면서 윈도우를 적용하여 이웃에 들어오는 코너점들을 모두 제거하는 방식으로 인접되어 검출된 코너점들을 특징점에서 제거하였다. 위에서 설명한 특징점 추출방법을 알고리즘으로 정리하면 다음과 같다.

단계 1: 영상 $I(x, y)$ 에 가우시안 필터를 적용하여 $\hat{I}(x, y)$ 를 구한다.

단계 2: 영상 $\hat{I}(x, y)$ 의 영상 그래디언트 $\hat{I}_x(x, y), \hat{I}_y(x, y)$ 를 구한다.

단계 3: 영상 그래디언트의 크기(magnitude)를 이용하여 에지 영상 $G(x, y)$ 를 구한다.

$$G(x, y) = 1 \text{ if } \sqrt{\hat{I}_x^2(x, y) + \hat{I}_y^2(x, y)} < edgeTh$$

$$= 0 \text{ otherwise}$$

단계 4: 에지($G(x, y) = 1$)인 각 화소 (x, y) 에 대하여,
 4.1: $(2W + 1) \times (2W + 1)$ 윈도우 크기의 이웃 화소들의 이미지 그래디언트를 이용하여 식 (1)의 공분산 행렬, C 을 구한다.
 4.2: 식 (3)에 의해 코너점 정도를 계산하고,
 $R_2(x, y) > cornerTh$ 인 (x, y) 를 코너점으로 추출하기 위하여, (x, y) 와 $R_2(x, y)$ 를 함께, $R_2(x, y)$ 에 의해 내림차순으로 삼입 정렬된 리스트에 추가한다.

단계 5: $R_2(x, y)$ 에 의해 내림차순 정렬된 리스트를 조사해 가며, 각 점 (x, y) 의 이웃에 있는 점들을 리스트에서 삭제한다.

(알고리즘) 특징점 검출

3.2 대응점 후보 추출

기준영상과 탐색영상의 특징점 사이의 대응점을 구하기 전에 기준영상의 각 특징점에 대한 대응되는 후보점을 탐색영상의 특징점에 중에서 구한다.

기준영상의 각 특징점 $p_i \in P$ 에 대해, 탐색영상의 특징점 $q_j \in Q$ 들 중에 p_i 와 q_j 사이의 거리가 임계치($distTh$)보다 작은 특징점의 정규상관계수를 구하여 일정한 임계치(예, $corrTh = 0.8$) 이상인 탐색영상에서의 특징점들을 대응점 후보(candidate points), $s_{ik} \in S_i, S_i \subset Q$ 로 구한다. 첨자 k 는 S_i 에 있는 후보점들을 구별하기 위한 첨자이고 S_i 의 원소 개수, 즉 p_i 의 대응점 후보 개수는 d_{ik} 이다. 대응점 후보가 없는($count(S_i) = 0$, 또는 $d_{ik} = 0$) 기준영상의 특징점 p_i 는 P 에서 제거한다.

각 특징점 사이의 정규상관계수 만을 사용하여도 대부분의 경우 후보점들 중에 원하는 대응점이 포함된다. 그러나 제한된 윈도우를 사용한 정규상관계수이기 때문에 기준영상의 특징점에 대한 탐색영상의 대응점이 없는 경우에도 높은 정규상관계수를 갖는 후보점이 포함될 수 있다. 정규

상관계수의 임계치를 너무 작게하면 실제로는 대응점이 없는 특징인데도 불구하고 후보점을 가질 수 있으며, 정규상관계수의 임계치를 너무 크게하면 실제 대응점이 후보점에 포함되지 않을 수 있다.

본 논문에서는 기준영상과 탐색영상의 각 특징점의 4방향에서의 이웃노드들과의 평균거리를 개별적으로 구하여, 두 정합되는 특징점 사이의 각 방향의 평균거리의 차이가 작을 것으로 가정한다. 기준영상의 특징점 p_i 의 x 좌표 보다 작은 이웃 특징점(서쪽), $N_W(p_i)$ 들의 평균거리 $D_{1,W}(p_i)$ 와, 특징점 p_i 의 x 좌표 보다 큰 이웃 특징점(동쪽), $N_E(p_i)$ 들의 평균거리 $D_{1,E}(p_i)$ 를 식 (10)와 같이 정의하며, x 좌표를 기준으로 $D_{1,N}(p_i), D_{1,S}(p_i)$ 를 정의한다. 또한 탐색영상의 특징점 q_i 에 대한 이웃 특징점의 4방향 평균거리 $D_{2,W}(q_i), D_{2,E}(q_i), D_{2,N}(q_i), D_{2,S}(q_i)$ 도 유사하게 정의된다. 이때 정규상관계수에 의해 구한 후보점에 한번도 포함되지 않은 특징점은 제외한다.

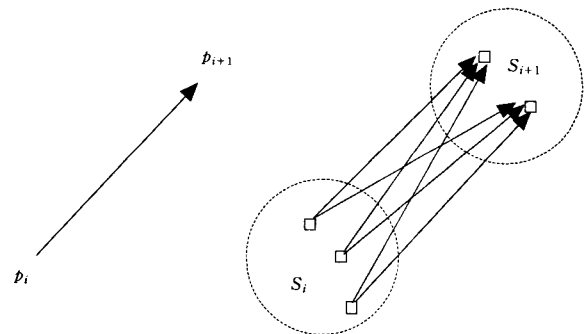
$$D_{1,W}(p_i) = \frac{1}{count(N_W(p_i))} \sum_{r_k \in N_W(p_i)} dist(p_i, r_k) \quad (10)$$

$$D_{1,E}(p_i) = \frac{1}{count(N_E(p_i))} \sum_{r_k \in N_E(p_i)} dist(p_i, r_k)$$

임계치 이상의 정규상관계수에 의해 구한 기준영상의 각 특징점 $p_i \in P$ 에 대한 대응점 후보 $s_{ik} \in S_i$ 들 사이의 각 방향의 평균거리차이가 차이가 임계치 보다 작은 대응점 후보가 없는 p_i 는 P 에서 제거한다.

3.3 동적 프로그래밍에 의한 대응점 정합

본 논문에서는 기준영상의 특징점 $p_i \in P, i = 0, \dots, n-1$ 를 연결된 그래프로 생각하고, 이에 대한 대응점 후보 집합 $s_{ik} \in S_i, i = 0, \dots, n-1, k = 0, \dots, d_i-1$ 에서 기준 영상의 특징점 p_i 에 의한 그래프와 유사한 대응점 후보를 선택하는 방법으로 대응점 정합을 한다.



(그림 1) 기준영상의 인접한 특징점 p_i, p_{i+1} 과 대응점 후보집합 S_i, S_{i+1}

두 그래프의 유사성은 인접한 특징점 사이의 거리, 방향, 이웃노드와의 평균거리를 고려하여 다음의 에너지 식으로 표현한다.

$$E_{total} = E(p_0, p_1, \dots, p_{n-1}; s_0, s_1, \dots, s_{n-1})$$

$$= E_0(p_0, p_1; s_0, s_1) + E_1(p_1, p_2; s_1, s_2)$$

$$+ \dots + E_{n-1}(p_{n-1}, p_n; s_{n-1}, s_n) \quad (11)$$

여기서 $E_i(p_i, p_{i+1}; s_i, s_{i+1}) = D_{iA} + wD_{iB}$

$$D_{iA} = |dist(p_i, p_{i+1}) - dist(s_i, s_{i+1})|$$

$$D_{iB} = |D_{1,w}(p_i) - D_{2,w}(s_i)| + |D_{1,w}(p_{i+1}) - D_{2,w}(s_{i+1})|$$

$$+ |D_{1,E}(p_i) - D_{2,E}(s_i)| + |D_{1,E}(p_{i+1}) - D_{2,E}(s_{i+1})| \quad (12)$$

$$+ |D_{1,N}(p_i) - D_{2,N}(s_i)| + |D_{1,N}(p_{i+1}) - D_{2,N}(s_{i+1})|$$

$$+ |D_{1,S}(p_i) - D_{2,S}(s_i)| + |D_{1,S}(p_{i+1}) - D_{2,S}(s_{i+1})|$$

기준영상의 특징점 p_i 에 대한 후보점은 집합 S_i 이며, 후보점의 수는 d_i 개이다. S_i 에 있는 각 후보점을 구분하기 위한 첨자 k 는 식을 단순화하기 위해서 생략하였으며, $s_i \in S_i$ 이다. 순환 그래프를 위해 $p_n = p_0$ 이고, $S_n = S_0$ 이다. D_{iA} 는 $\overline{p_i p_{i+1}}$ 거리와 $\overline{s_i s_{i+1}}$ 거리의 차이가 작을수록 정합될 특징점으로 고려하기 위해서이며, D_{iB} 는 p_i, s_i 의 이웃노드의 평균 거리 차이와 p_{i+1}, s_{i+1} 의 이웃노드의 평균 거리차이가 작을수록 정합될 특징점으로 고려하기 위한 것이다. $\overline{p_i p_{i+1}}$ 와 $\overline{s_i s_{i+1}}$ 사이 각이 90도를 넘는 후보점들은 고려하지 않았다. D_{iA} 와 D_{iB} 사이의 가중치는 $0 \leq w \leq 1$ 이다. 대응점을 결정하는 것은 식 (11)을 최소화하는 s_i 를 결정하는 것이다. 하나 이상의 기준영상의 특징점이 하나의 탐색영상 특징점으로 대응되는 것은 이때 가장 큰 정규 상관계수를 갖는 기준영상의 특징점으로 결정한다. 제안된 동적 프로그래밍에 의한 대응점 매칭 방법을 알고리즘으로 정리하면 다음과 같다.

단계 1 : 정규상관계수를 이용하여 기준영상의 각 특징점 p_i 에 대한 탐색영상의 대응점 후보 $s_{ik} \in S_i$ 를 구하고, $count(S_i) = 0$ 인 p_i 를 기준영상의 특징점 집합 P 에서 제거한다.

단계 2 : 식 (10)에 의해 기준영상과 탐색영상의 특징점의 평균 이웃노드 거리를 계산한다. 기준영상의 각 p_i 와 대응점 후보, $s_{ik} \in S_i$ 사이의 차이가 임계치 보다 작은 후보점이 하나도 없는 p_i 를 기준영상의 특징점 집합 P 에서 제거한다.

단계 3 : 식 (11)를 동적 프로그래밍으로 최소화하기 위하여 에너지 행렬, energyMtx과 위치 행렬, posMtx이 필요하다. 기준영상의 각 특징점 p_i 에 대한 후보 특징점(candidate points)

// N : 대응점 후보들이 있는 기준영상의 특징점 개수
 // MAX : 기준영상의 각 특징점의 최대 후보점 개수
 입력 :
 Point P[N+1]; // 기준영상의 특징점을 저장하고 있는 배열

```

Point S[N+1][MAX]; // 기준영상의 후보점을 저장하고
                    // 있는 배열
int D[N+1]; // 후보점의 수를 저장하고 있는
            // 배열
출력 :
Point Q[N+1]; // 참조영상의 대응점을 저장할 배열

입시변수 및 배열 :
float energyMtx[N+1][MAX]; // 각 단계까지의 최소 에너
                            // 지 저장을 위한 실수 배열
int posMtx[N+1][MAX]; // 위치 저장을 위한 정수 배열
int i, j, k, m, min_position, min_final_position, pos;
float min_energy, energy, min_final_energy;
float dA, dB, w = 0.3;
Point nNode, cNode;
Point A, B;
float nA, nB, dot, angle;
// 에너지 행렬을 초기화한다.
for(i = 0; i < D[0]; i++) energyMtx[0][i] = 0.0;
for(i = 0; i < N; i++) // stage
{
    // if i+1 >= N, i+1 = 0
    p1 = P[i];
    p2 = P[i+1];
    for(j = 0; j < D[i+1]; j++) // next
    {
        min_energy = BIG;
        nNode = S[i+1][j];
        for(k = 0; k < D[i]; k++) // current
        {
            cNode = S[i][k];
            // 각도 계산
            A.x = p2.x - p1.x; A.y = p2.y - p1.y;
            B.x = nNode.x - cNode.x;
            B.y = nNode.y - cNode.y;
            nA = sqrt(A.x * A.x + A.y * A.y);
            nB = sqrt(B.x * B.x + B.y * B.y);

            dot = (A.x * B.x + A.y * B.y);
            angle = dot / (nA * nB);
            if(angle < 0.0) // 90도 보다 크면 고려하
                            // 지 않는다.
                energy = BIG;
            else
                energy = energyMtx[i-1][k] + dA
                    + w * dB;
        }
        if(energy < min_energy)
        {
            min_energy = energy;
            min_position = k;
        }
    }
}
    
```

```

        energyMtx [i][j] = min_energy ;
        posMtx [i][j] = min_position ;
    }
}
// 에너지 배열에서 에너지가 최소가 되는 마지막 컬럼 위치를
// 찾는다.
min_final_energy = BIG ;
for(m = 0 ; m < D[N - 1] ; m++){
    if(energyMtx[N-1][m] < min_final_energy)
    {
        min_final_energy = energyMtx[nOfSnaixel - 1][m] ;
        min_final_position = m ;
    }
}
// 위치 행렬을 이용하여 역으로 에너지가 최소가 되는 대응
// 점 후보들을 구한다.
pos = min_final_position ;
for(i = N - 1 ; i > 0 ; i--){
    Q[i] = S[i][pos] ;
    pos = posMtx [i-1][pos] ;
}
Q[0] = S[0][pos] ;
}
    
```

(알고리즘) 동적 프로그래밍에 의한 대응점 매칭

4. 구현 및 실험결과

본 논문에서 제안된 알고리즘을 적용하여 두 영상에서 에지를 검출하고, 검출된 에지에서 코너점 정도를 계산하여 코너 특징점을 추출하고, 추출된 특징점 사이의 정합되는 특징점을 동적 프로그래밍을 사용하여 대응되는 특징점을 정합하는 실험을 하였다.

(그림 2)는 코너점 검출 알고리즘을 250×250 크기의 house 스테레오 영상에 *edgeTh* = 30, *cornerTh* = 0.5를 적용하여, $R_2(x, y) > 0.5$ 보다 큰 코너점들을 구하고, $W = 15$ 의 윈도우를 사용하여 인접한 코너점들 중에서 큰 값의 코너점 정도를 갖는 코너점만을 추출한 결과이다. (그림 2)에서 두 영상에서의 특징점의 번호는 코너점 정도가 큰 순서로 같은

번호 사이에 대응 관계는 아니다.



(a) 기준영상 (b) 탐색영상

(그림 2) house 영상

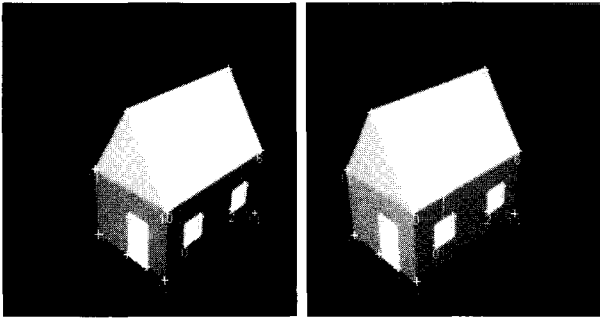
<표 1>은 (그림 2)의 기준영상(왼쪽)의 특징점에 대한 탐색영상(오른쪽)의 특징점들 중에서 대응점 후보들을 *corrTh* = 0.8로 하여 구한 결과이다. 예를 들어 기준영상의 0(q_0)번 특징점에 대한 대응되는 특징점 후보집합 S_0 는 탐색영상의 0(q_0)번 특징점과 3(q_3)번 특징점이며, 탐색영상의 0번 특징점과의 정규상관계수는 $NCC(0, 0) = 0.9781$ 이며, 탐색영상의 3번 특징점과의 정규상관계수는 $NCC(0, 3) = 0.9905$ 이다. 그러므로 정규상관계수가 가장 높은 특징점을 정합되는 대응점으로 하면 기준영상의 0번 특징점과 탐색영상의 3번 특징점이 정합되는 대응점으로 구하게 된다. S_1 을 보면 기준영상의 1번 특징점에 대한 탐색영상의 대응점 후보는 1, 2번 특징점이며 정규상관계수는 2번 특징점이 높기 때문에 잘못 정합될 수 있다. 또한 S_5 와 S_{13} 을 보면 대응점 후보가 탐색영상의 5, 14번 특징점으로 같은 것을 알 수 있으며, S_{12} 를 보면 기준영상의 12번 특징점에 대한 후보 특징점이 없으므로 기준영상의 특징점에서 제거되어, (그림 3)(a)와 같이 된다.

(그림 3)(b)는 제안된 동적 프로그래밍 정합 방법을 적용하여 (그림 3)(a)의 기준 영상의 특징점에 대한 탐색영상의 대응점 후보들 중에서 에너지를 최소화하는 특징점을 선택

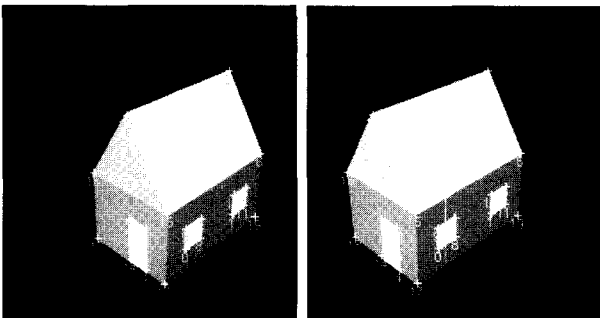
<표 1> 대응점 후보 S_i

대응점 후보							
S_0	S_1	S_2	S_3	S_4	S_5	S_6	S_7
0(0.9781) 3(0.9905)	1(0.9786) 2(0.9671)	3(0.9806)	4(0.9979)	1(0.9563)	5(0.9872) 14(0.8926)	6(0.9962)	7(0.9584)
대응점 후보							
S_8	S_9	S_{10}	S_{11}	S_{12}	S_{13}	S_{14}	S_{15}
11(0.9918)	8(0.9869)	12(0.9811)	13(0.9817)		5(0.8676) 14(0.9845)	11(0.9158) 15(0.9938)	16(0.9629)

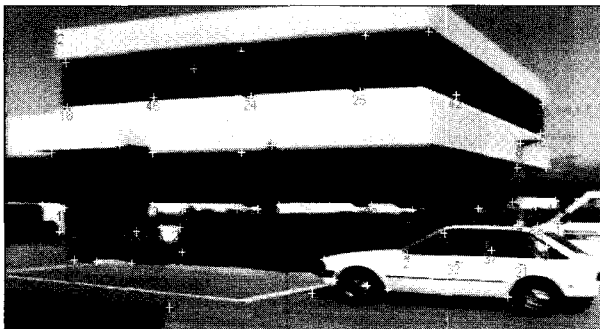
하여 정합한 결과이다.



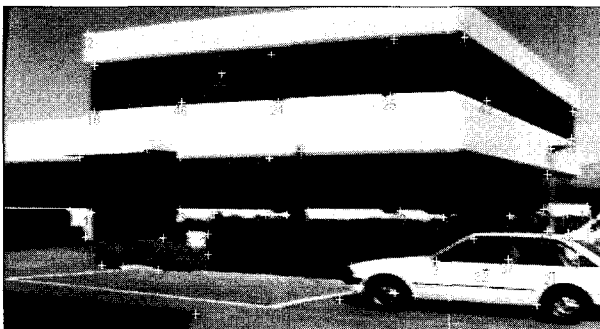
(a) 기준영상 (b) 탐색영상
(그림 3) house 영상의 대응되는 특징점 검출 ($W = 15$)



(a) 기준영상 (b) 탐색영상
(그림 4) house 영상의 대응되는 특징점 검출 ($W = 7$)



(a) 기준영상



(b) 탐색영상

(그림 5) building 영상의 대응되는 특징점 검출

(그림 4)는 house 스테레오 영상에서 윈도우 크기를 $W = 7$ 로 하여 코너점을 추출하고, 동적 프로그래밍을 적용한 결과로 모든 특징점이 올바르게 정합되었다. 정규상관계수를 적용하여 대응점을 구했을 때는 2개의 대응점에서 잘못 정합되었다.

(그림 5)는 $edgeTh = 30$, $cornerTh = 1$, $W = 15$ 를 적용하여 기준영상에서 69개의 특징점을 추출하고, 탐색영상에서 68개의 특징점을 추출하였다. (그림 5)(a)는 정규상관 계수 임계치 $corrTh = 0.8$ 를 적용하여 기준영상의 특징점에 대한 탐색영상의 후보점을 추출하고 후보점이 없는 특징점을 제거한 그림이고, (그림 5)(b)는 이에 대응하는 탐색 영상에서의 58개의 특징점이다. 정규상관계수만을 적용한 실험에서도 잘못 검출된 대응점이 없었다. 이것은 검출된 특징점에서의 영상 밝기 특징이 뚜렷하였기 때문이다.

5. 결 론

본 논문에서는 기준영상과 탐색영상 사이의 대응되는 특징점을 정합하는 알고리즘을 제안하였다. Harris 코너점 검출기에 의해 특징점을 검출하고, 기준영상의 각 특징점에 대해, 정규상관계수가 임계치 이상인 탐색영상의 특징점으로 후보 정합점을 구한 다음, 동적 프로그래밍을 사용하여 후보 정합점들 중에서 대응되는 특징점을 효율적으로 구하였다. 실험 결과 밝기변화가 많은 실외 영상에서는 대부분의 경우 정규상관계수만을 사용할 때도 만족할 만한 결과를 얻었으나 특징 변화가 적은 곳에서는 정규상관계수만을 사용할 때는 잘못 정합되는 경우가 발생하는 것을 알 수 있었다. 향후연구로는 부 화소단위의 정합과 에피폴라 기하를 적용한 정합에 대한 연구가 필요하다.

참 고 문 헌

- [1] Z. Zhang, R. Deriche, O. Faugeras, Q. T. Luong, "A Robust Technique for Matching Two Uncalibrated Images Through the Recovery of the Unknown Epipolar Geometry," INRIA Technical Report, 1994.
- [2] Z. Zhang, "Determining the Epipolar Geometry and its Uncertainty : A Review," INRIA Technical Report, 1996.
- [3] S. M. Smith and J. M. Brady, "SUSAN-A New Approach to Low Level Image Processing," TR95SMS1c, Oxford University, UK.
- [4] E. Trucco, A. Verri, Introductory Techniques for 3-D Computer Vision, Prentice Hall, 1998.
- [5] D. H. Ballard and C. M. Brown, Computer Vision, Prentice Hall, Inc, 1982.
- [6] B. S. Manjunath and C. Shekhar, and R. Chellappa, "A New Approach to Image Feature Detection with Application,"

[7] N. Merlet and J. Zerubia, "New Prospects in Line Detection by Dynamic Programming," IEEE PAMI, Vol.18, No.4, pp. 426-431, 1996.

[8] J. C. Clarke, Application of Sequence Geometry to Visual Motion, PHD Thesis, Robotics Research Group, Dept. of Engineering Science, University of Oxford, 1997.

[9] C. Tomasi and T. Kanade, "Shape and Motion from Image Streams : a Factorization Method-3," Detection and Tracking of Point Features, Technical Report CMU-CS-91-132, Carnegie Mellon University, Pittsburgh, 1991.



김 동 근

e-mail : kdk@cntc.ac.kr

1989년 충남대학교 계산통계학과(이학사)

1991년 충남대학교 계산통계학과(이학석사)

1996년 충남대학교 전산학과(이학박사)

1996년~2001년 서남대학교 조교수

2002년~현재 천안공업대학 컴퓨터과 전임강사

관심분야 : 영상처리, 컴퓨터 비전, 멀티미디어 응용, 컴퓨터 그래픽스