

Temporal Texture Modeling for Video Retrieval

金 度 年* · 趙 東 燮**
(Do-Nyun Kim · Dong-Sub Cho)

Abstract - In the video retrieval system, visual clues of still images and motion information of video are employed as feature vectors. We generate the temporal textures to express the motion information whose properties are simple expression, easy to compute. We make those temporal textures of wavelet coefficients to express motion information, M components. Then, temporal texture feature vectors are extracted using spatial texture feature vectors, i.e. spatial gray-level dependence. Also, motion amount and motion centroid are computed from temporal textures. Motion trajectories provide the most important information for expressing the motion property. In our modeling system, we can extract the main motion trajectory from the temporal textures.

Key Words : Temporal texture, wavelet transform, texture features, motion, trajectory

1. 서 론

통신과 멀티미디어 기술이 급속하게 발전하면서 디지털 라이브러리에서 방대한 양의 시각 데이터에 접근하는 것이 현실화되고 있으며 이러한 시각 데이터에 필요한 시간 안에 필요한 만큼의 정보를 얻기 위해서는 동영상 검색 시스템이 필요하다. 동영상을 검색하기 위해서는 기존의 데이터베이스 검색 시스템에서 사용했던 검색기가 필요하며 동영상에서는 기존의 문서 데이터베이스 검색에서 사용되었던 키워드와 색인 등과 같은 동영상에 따른 색인이 필요하게 되었다. 동영상 검색을 위한 색인 시스템은 문서 검색에 적용되는 색인 시스템과는 달리 다양한 특성을 필요로 한다. 그 중 중요한 정보가 동영상에서의 움직임 정보이다. 왜냐하면 동영상 검색 시스템은 정지 영상에 대해 개발된 영상 검색 시스템의 모든 요소를 이용하면서 동영상에만 존재하는 정보, 즉 움직임 정보가 추가로 존재하기 때문이다.

동영상 데이터에서 추출될 수 있는 많은 시각적인 정보는 이미 기존의 내용 기반 검색 시스템에서 많이 연구되어 왔지만 아직도 연구가 진행되고 있고 앞으로도 많은 성과가 있어야 하는 부분이 움직임 정보를 자동으로 추출하는 작업이다. 인간의 인지 과정을 거친 의미론적인 움직임 정보가 가장 이상적이긴 하지만 현실적으로 인간의 인지 과정은 주관적이고 또한 처리해야 할 동영상 데이터의 양이 매우 방대하므로 인간의 인지 과정을 빌린 고 수준 의미 정보 시스템은 불가능

하다. 따라서 움직임 정보를 추출하는 방법이 자동화되어야 한다. 이러한 자동화를 이루기 위해서는 움직임을 정량적으로 표현할 수 있는 표현 수단과 크기가 있어야 한다. 이러한 움직임 주석의 방법으로 템포럴 텍스처(temporal texture)를 이용한 움직임 표현 방법을 제안하였다.

2. 관련 연구

문서 검색에 머물러 있던 정보 검색이 컴퓨터 하드웨어의 발전과 더불어 시각적인 정보의 검색으로 그 용도를 넓히게 되었다. 시각적인 데이터에서 사용자가 필요로 하는 정보를 추출하기 위해서는 여러 분야의 연구가 수행되어야 한다. 첫 번째, 시각적인 저수준 데이터 혹은 동영상에서 오브젝트를 자동으로 분할하고 추적해야 하고 두 번째, 풍부한 시각 특성 라이브러리를 구축하여야 하고 셋째, 원하는 오브젝트를 표현할 수 있는 질의어 시스템, 넷째, 월드 와이드 웹을 통해 대화식으로 질의하고 브라우징할 수 있는 시스템에 관한 연구가 필요하다. 동영상 검색에 사용될 움직임 정보를 모델링하기 위한 방법으로 템포럴 텍스처를 생성하였다. 관련 연구로는 움직임 정보 모델링에 사용된 템포럴 텍스처에 관한 기존의 연구 중 Polana[1], Szummer[2], Liu[3]의 연구를 중심으로 고찰하였다.

세 가지 접근 방법의 차이는 우선 템포럴 텍스처의 정의의 차이에서 기인한다. Polana의 경우 움직임 정보를 세 가지 각각 다른 유형으로 정의하여 각각에 다른 처리 방법을 제안하였으며 이 세 가지 유형의 움직임 중 템포럴 텍스처와 반복 운동만을 연구의 대상으로 하였다. 템포럴 텍스처의 정의는 통계적 규칙성을 보이지만 공간적으로 불확실한 움직임 패턴으로 하였으며 이 정의를 Liu가 특히 입의 발생에 초점을 맞춰 다시 사용하였다. Szummer는 정의의 범위를 축소하

* 正 會 員 : LG 電 子 技 術 員 · 工 博

** 終 身 會 員 : 梨 花 女 子 大 學 計 算 機 學 科 教 授 · 工 博

接 受 日 字 : 2000 年 11 月 28 日

最 終 完 了 : 2001 年 3 月 5 日

여 공간 텍스처가 움직이는 것으로 정의하였다.

이러한 정의에 따라 Polana는 템포럴 텍스처의 경우 기존의 공간 텍스처에 사용된 특성 벡터들을 통계적이고 주파수 도메인에서 기본 흐름(normal flow)을 이용하여 모델링하였으며 반복 운동의 경우 정의에 의해 주기성을 가지고 있으므로 주기성을 탐색하는 과정을 연구하였다. Szummer는 연속 예측을 표현할 수 있는 여러 모델 중 STAR 모델을 이용하여 움직이는 텍스처를 모델링하였다. 그러나 이 모델링은 선형성을 가정하고 있으므로 회진, 확장, 주기적인 반복 운동 등은 모델링할 수 없다는 단점이 있다. Liu는 Wold 분할을 이용하여 움직임을 모델링하였으며 특히 움직임을 주기성을 탐색하는데 주력하였다. 표1에서는 이러한 세 모델들이 사용한 정의와 모델링 기법을 비교하였다[8].

표 1 각 모델간의 템포럴 텍스처 정의와 특징 비교
Table 1 Comparison of definition of temporal texture and modeling properties

모델명	템포럴 텍스처의 정의	모델링 기법의 특징
Polana	템포럴 텍스처의 정의는 통계적 규칙성을 보이지만 공간적으로 불확실한 움직임 패턴	· 움직임을 종류를 3가지로 구분하여 각각 다른 방법을 적용 · 템포럴 텍스처 - 광류와 통계치 이용 · 반복 운동 - 주기성 검색
Szummer	움직임이 있는 텍스처	STAR 이용
Liu	Polana의 템포럴 텍스처 정의를 이용, 특히 임의 발생 강조	Wold 분할 이용

3. 템포럴 텍스처 생성 기법

움직임 인식 시스템은 여러 중요한 영역에서 많은 잠재력 있는 응용 분야를 갖고 있다. 많은 자연 객체들의 움직임은 주기적 운동으로 분류될 수 있다. 많은 기계시스템에서 이러한 움직임의 인식 능력은 자동화된 감시 시스템 같은 응용에서 유용하다. 또한 움직임 인식에 기반한 입술 움직임 탐지 시스템은 화자의 입술의 움직임과 상이한 발생에 대한 입술 위 영역의 움직임을 특성화함에 의해 발생의 인식 비율을 개선하는 데 사용될 수도 있다. 손 동작 인식 시스템은 더 나은 사람과 기계간의 인터페이스를 설계하는데 도움을 줄 수도 있다. 이러한 움직임 인식 시스템은 움직임의 형태를 분석함으로써 구현될 수 있다. 특히 템포럴 텍스처를 이용한 움직임 인식은 오브젝트 중심의 움직임 인식보다 움직임에 중점을 두어 분석할 수 있다는 장점이 있으며 그 외에도 배경 소거의 문제나 오브젝트 구별(identification) 문제 등 실시간에 수행하기 어려운 문제를 해결하는데도 도움이 된다. 이 장에서는 이러한 움직임 인식 및 모델링에 사용될 템포럴 텍스처를 생성하는 방법을 제안하였다.

3.1 웨이브릿 변환 계산 기법

고속 푸리에 변환(FFT)과 같이 웨이브릿 변환은 정수를

지수로 갖는 2의 멱수 길이의 벡터에서 빠른 선형 연산(linear operation)을 한다. 또한 FFT와 같이 역변환가능하고 orthogonal하다. 역변환을 큰 행렬로 간주하면 변환의 전치 행렬로 생각하면 된다[4].

본 연구에서 사용된 템포럴 웨이브릿(temporal wavelet)은 2차원 웨이브릿이 공간 데이터 간의 중복성을 없애기 위해 수행된다면 템포럴 웨이브릿은 시간상의 중복을 제거하기 위해 수행된다.

3.2 템포럴 텍스처 정의

템포럴 텍스처를 움직임 분류에 필요한 특성으로 사용하기 위해서는 정의를 확장할 필요가 있다. 그 이유는 동영상에서는 Polana가 제안한 것처럼 템포럴 텍스처, 반복 운동, 특이 운동 등으로 미리 나누어 움직임을 인식하는 것은 세 클래스를 자동적으로 분류해 주는 특정 척도가 제안되지 않은 상태에서는 비현실적이고 Szummer가 제안한 것처럼 단순히 움직이는 텍스처로 의미를 축소하기에는 동영상 분류라는 입장에서 얻을 수 있는 득이 거의 없으며 수학적으로 강인하다는 Liu의 동질(homogeneous)의 환경이라는 정의는 지나치게 모호하다. 따라서 이러한 연구를 진행하기에 앞서 템포럴 텍스처의 정의를 확장하는 것이 필요하다. 본 연구에서 제안한 템포럴 텍스처는 다음과 같다.

용어 정의1 템포럴 텍스처 (temporal texture)

임의의 시간 간격내의 움직임으로부터 생성되었으며 지역적으로는 통계적 유사성을 보이는 패턴

템포럴 텍스처를 웨이브릿으로 압축된 동영상으로부터 생성하기 위하여 움직임 정보를 포함한 웨이브릿 계수 열을 선택하여야 한다. 이때 사용되는 웨이브릿 계수 열을 M 요소라고 하고 이 M 요소는 계수 열 중에서 저주파 정보를 포함하는 S 요소에서 선택한다.

용어 정의2 M 요소 (motion component)

웨이브릿 변환을 한 결과로서 생성된 계수 열들 중 S 요소의 고주파 정보

M 요소에 대한 자세한 설명은 다음과 같다. 웨이브릿 변환을 하게 되면 서로 다른 단계의 다중 해상도를 가진 부대역을 얻을 수 있다. 2차원 영상의 경우 부대역은 저주파 성분(LL)과 고주파 성분(LH,HL,HH)으로 나뉘게 되며, LL 대역의 영상은 이전 영상의 1/2의 해상도를 가진 얇은 형태가 된다(그림 1). 이를 매끈한 성분(smooth component), 즉 S 요소라고 부른다. 또한, LH,HL,HH 대역은 각각 수직, 수평, 대각선 방향의 고주파 성분을 가진다. 이를 세부 성분(detail component), D 요소라고 부른다. 웨이브릿으로 변환된 각 대역은 변환 단계 L과 방향(LL,LH,HL,HH)을 나타내는 O를 이용하여 (L,O)의 형태로 표현될 수 있다. 각 대역은 원 영상과 닮은 형태를 갖게 된다. 즉, S 요소에 영상의 중요한 정보가 거의 들어가게 된다. 웨이브릿 변환된 동영상에서 영상의

중요한 정보가 포함된 S 요소를 선택하면 그 S 요소의 D 요소에 움직임 정보가 있다. 이러한 S 요소의 D 요소를 움직임 성분이라는 의미로 M 요소(motion component)라는 용어를 정의하였다.

L=2 O=LL	L=2 O=LH	L=1 O=LH
L=2 O=HL	L=2 O=HH	
L=1 O=HL		L=1 O=HH

그림 1 2차원 영상의 웨이브릿 계수
Fig. 1 Wavelet coefficients of 2D images

3.3 템포럴 텍스처 생성

동영상의 템포럴 텍스처 특성을 알아내기 위해서는 대상 동영상의 템포럴 텍스처를 만들어야 한다. 움직임 특성을 분석하기 위한 템포럴 텍스처는 분석 대상이 된 동영상의 움직임 정보를 표현해야 한다. 현재 사용되는 영상 데이터는 수년 전의 영상 데이터보다 훨씬 데이터 양이 크기 때문에 압축 영상에서 직접 만들어 내려 하였다. 대상으로 한 압축 방법은 웨이브릿 변환을 이용한 압축 방법이며 템포럴한 특성을 압축하므로 3차원 변환과는 의미 상 차이를 보인다. 알고리즘 1은 이러한 웨이브릿 계수로부터 템포럴 텍스처를 생성하는 알고리즘이다[5].

알고리즘 1 템포럴 텍스처 생성

- 단계 1 : 동영상에 대하여 웨이브릿 변환하고, 혹은 이미 변환된 결과에 대해 S 요소를 선택한다.
- 단계 2 : 분할 규칙에 따라 S 요소를 선택하여 이 중에서 고주파 성분, 즉 M 성분을 선택한다.
- 단계 3 : 단계 2에서 구해진 성분을 가중 결합(weighted sum)하여 한 장의 영상으로 만든다.

웨이브릿 변환의 결과로 만들어진 이미지 열에서 동영상을 분류하고 템포럴 텍스처를 만들 때는 움직임 특성이 많이 반영된 부분을 선택하는 것이 중요하다. 따라서 이러한 부분을 선택하는 규칙이 필요하게 된다. 이 규칙을 본 연구에서는 분할 규칙이라 하고 이 분할 규칙은 움직임 특성을 많이 포함한 부분을 추출해 내는 우선 순위에 관한 규칙이다. 동영상에서 일정한 간격으로 n번 샘플링하여 n개의 영상을 추출한다. 이렇게 추출된 n의 영상을 웨이브릿 변환하여 그림 2와 같은 웨이브릿 계수 열을 구한다. 동영상의 해상도가 i*이 일 때 웨이브릿 계수 열은 동영상의 각 프레임의 크기와 같은 i*이인 배열 n개로 구성된다. 여기서 움직임 정보가 모여있는 M 요소를 선택하는 방법이 알고리즘2 이다[6].

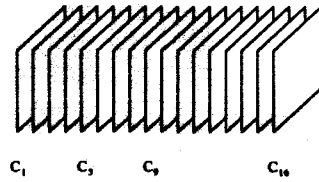


그림 2 웨이브릿 계수 열
Fig. 2 Sequences of wavelet coefficients

알고리즘 2 분할 규칙(Division Rule)

C_k 는 $i*이인 배열, k=1, \dots, n$

$n=2^m$, m 은 $m \geq 2$ 인 정수

$$S_N = \{k | \forall C_k, k=1, \dots, 2^{N+2}\}$$

$$M_N = \{k | \forall C_k, k=(2^{N+1}+1), \dots, 2^{N+2}\}$$

여기서 $N=0, \dots, \log_2 n - 2$

$n=2^m$ 인 이유는 2의 멱수일 때 웨이브릿 변환이 효율적이기 때문이다. 따라서 $i \geq 2$ 의 멱수일 때 효율적이 된다. 이 알고리즘에 의하면 S_0 의 경우에는 C_1, C_2, C_3, C_4 가 선택되며 이중 고주파 성분, 즉 D 요소인 C_3, C_4 가 M_0 가 된다.

그림 3은 실험에 사용한 동영상이며 그림 4에서는 분할 규칙에 의해 S 요소와 M 요소를 선택할 때 $N=0$ 인 경우의 결과 계수를 역변환하여 보인 것이며 그림 5는 $N=1$ 인 경우이다.



그림 3 실험 동영상의 예
Fig. 3 Video example

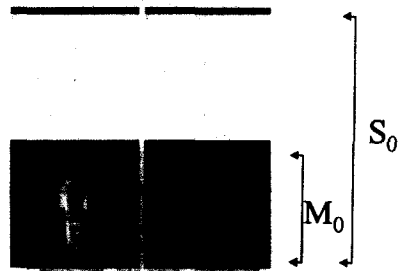


그림 4 각 요소 선택 (N=0의 경우)
Fig.4 Results of Division Rule (N=0)

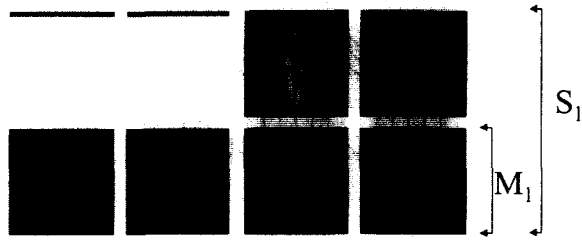


그림 5 각 요소 선택 (N=1의 경우)
Fig. 5 Results of Division Rule (N=1)

그림 5에서 선택한 S 요소 중 고주파 부분에 움직임 정보가 모여 있으므로 고주파 부분을 추출하여 필요한 경우에는 가중치 결합 연산을 거쳐 텍스처 영상으로 변환한다. 결과를 그림 6에 보였다.



그림 6 그림 3의 동영상에 대한 텍포럴 텍스처
Fig. 6 Temporal texture for Fig. 3

그림 7은 또 다른 동영상의 예이며 등간격으로 16장을 샘플링하였다. 이 동영상은 화면의 왼쪽에서 사람이 뛰어 들어와 오른쪽으로 이동한다. 그림 8에는 이 예에 웨이브릿 변환한 후의 결과를 역변환하여 보였으며 그림 9에는 선택된 M 요소의 예를 보였다. 이 M 요소부터 가중 결합되어 생성된 텍포럴 텍스처를 그림 10에 보였다.



그림 7 동영상의 다른 예
Fig. 7 Example

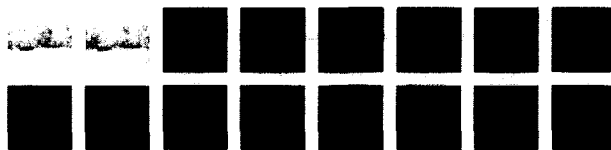


그림 8 그림 7의 웨이브릿 변환 결과
Fig. 8 Results of wavelet transform



그림 9 그림 8에서 선택한 M1 요소
Fig. 9 M1 components of Fig. 8



그림 10 생성된 텍포럴 텍스처
Fig. 10 Temporal texture

3.4 제안한 동영상 텍포럴 텍스처와 기존의 텍포럴 텍스처와의 비교

기존의 텍포럴 텍스처와 비교하여 제안한 텍포럴 텍스처를 비교하면 다음과 같다.

텍포럴 텍스처와 반복 운동을 연구한 Polana[1]의 연구에서 움직임을 세 부류로 나누어 각기 정의하였다는 것은 큰 성과이나 움직임을 미리 분류하여 각기 다른 방법으로 움직임을 인식하는 기법은 약점이기도 하다. Polana의 연구에서는 움직임의 종류가 세가지 있다는 것은 정의하였는데 움직임 종류가 동영상 데이터베이스에 섞여 있는 경우 어떻게 세 종류로 자동으로 분류할 것인지 분류 기준을 제시하지 못하였다. 그러나 이 연구에서는 움직임의 분류되어 있다고 가정하고 각기 다른 움직임 인식 기법을 적용하므로 실제로 자동으로 움직임의 부류를 결정할 척도가 없다면 동영상 색인 과정이 자동화되어야 한다는 본 연구의 필요성에 위배된다. 또한 이 연구의 또 다른 약점은 실제 동영상에 빈번하게 출현하는 특이 운동에 대한 내용이 연구되어 있지 않다. 이러한 약점에 비추어 볼 때 본 연구에서 제안한 텍포럴 텍스처는 움직임의 부류를 미리 나누지 않고 Polana의 세 종류의 움직임 부류에 동일한 기법을 적용하여 모델링할 수 있다는 것이 큰 장점이다. 동일한 기법을 이용하여 텍포럴 텍스처가 생성된 후에는 이 텍포럴 텍스처의 특성에 의해 움직임의 부류 및 움직임의 특징이 결정된다. 이 연구와 비교하여 제안한 텍포럴 텍스처의 단점은 반복 운동을 자세하게 구별해 내지 못한다는 것이다.

STAR를 이용한 텍포럴 텍스처의 모델링 연구와 비교하면 본 연구에서 제안한 텍포럴 텍스처의 장점을 더욱 쉽게 알 수 있다. Szummer[2]가 제안한 텍포럴 텍스처 모델링 연구에서는 텍포럴 텍스처가 단순히 공간 텍스처가 움직이는 것으로 정의하였다. 이 연구의 경우에는 Polana의 연구 중 텍포럴 텍스처 부분을 Polana의 연구에서는 광류를 이용한 반면 STAR라는 모델을 사용하여 모델링한 것이다. 광류를 사용한 것과의 뚜렷한 차이는 합성에도 동일한 모델링 기법을 사용할 수 있다는 것이다. 이 연구에서의 텍포럴 텍스처는 자연계의 움직이는 오브젝트 및 기계나 감시 시스템 등에서의 움직임 인식에 사용되기에는 제한이 많다. 또한 텍포럴 텍스처가 선형성을 가정한 경우에만 모델링할 수 있다는 제한도 시스템을 더욱 한정적이게 만든다. STAR를 이용한 텍포럴 텍스처와 본 연구에서 제안한 텍포럴 텍스처를 비교하면 제안한 텍포럴 텍스처는 선형성을 가정하지 않으며 특히 비선형성을 표현한다 하더라도 선형성을 가진 텍포럴 텍스처

보다 계산이 복잡해진다든지 하는 오버헤드가 전혀 없다. 또한 STAR 모델을 이용한 템포럴 텍스처가 모델링할 수 없는 회전, 가속 등의 움직임도 표현할 수 있다는 장점이 있다.

Wold 분할을 이용한 템포럴 텍스처 연구[3]는 움직임 인식에 Wold 분할을 이용하여 템포럴 텍스처를 모델링하여 사용하였다. 이 연구에서는 오브젝트를 추적하여 정렬하여 움직임을 보인 부분을 합성하여 움직임을 분석하였다. 이 시스템에서는 배경이 변화하는 것을 계속 감시하여야 하고 오브젝트를 추적해야 한다. 이 연구와 비교하여 본 연구에서 제안한 동영상 템포럴 텍스처는 움직임을 추적할 수 있는 능력이 있으면서도 배경 소거나 오브젝트 추적 등 많이 시간이 소요되는 연산들을 수행하지 않고도 움직임 정보를 정량화할 수 있다.

각 기존의 연구와 제안한 템포럴 텍스처 간의 기능의 비교는 표 2에 보였다.

표 2 템포럴 텍스처 기능 비교
Table 2 Comparison of functions

비교 항목		연구자	(1)	(2)	(3)	(4)
가능한 모델링	움직임 사전 분류 여부		필요	필요	필요	불필요
	템포럴 텍스처		가능	가능	불가능	가능
	반복운동		가능	불가능	가능	가능
	특이운동		불가능	불가능	가능	가능
배경 소거			필요	불필요	필요	불필요
오브젝트 추적 및 매칭			필요	불필요	필요	불필요
다중 오브젝트 처리 여부			가능	불가능	가능	불가능

(1)템포럴 텍스처와 반복 운동 연구 (2) STAR를 이용한 템포럴 텍스처 모델링 연구 (3) Wold 분할을 이용한 템포럴 텍스처 모델링 연구 (4) 본 논문에서 제안한 템포럴 텍스처

4. 템포럴 텍스처 특성 벡터 생성

움직임을 표현하는 템포럴 텍스처의 특성은 여러 각도에서 제안될 수 있으나 대략 2가지로 간추려 고려한다. 즉, 기존의 명도 텍스처에서 텍스처를 특징짓는 특성 벡터로서 사용된 공간적 명암도 의존성 중 움직임 정보를 표현할 수 있는 에너지, 엔트로피, 관성을 계산하여 한가지 특성으로 할당하고 두 번째 특성은 템포럴 텍스처에는 움직임 정보만 포함하고 있으므로 움직임의 양(amount)을 계산하고 그 움직임의 중심을 구하여 특징 벡터로 삼는 것이다. 많은 텍스처 특성벡터 중 에너지, 엔트로피, 관성 등 공간적 명암도 의존성 벡터들을 선택한 이유는 참고 문헌[7]에서 다른 기법들과 비교하여 분석시에 효과적이라고 보고되었기 때문이다.

명암도 i 와 j 인 두 개의 픽셀이 거리 d 만큼 떨어지고 방향이 θ 인 위치에 있다면 텍스처 특성은 i, j, d, θ 에 의존한다. 이러한 픽셀 쌍의 수를 P 라고 하면, P 는 $P(i, j, d, \theta)$ 로 나타낼 수 있다. 즉, 주어진 영상에서 $P(i, j, d, \theta)$ 는 명암도가 i

이고 j 이면서 이산적 거리 수치 d 만큼 떨어지고 이산적 각도 θ 만큼의 방향으로 나열된 픽셀 쌍들의 총 수이다[7].

용어 정의3 에너지

$$E(d, \theta) = \sum_{i=0}^L \sum_{j=0}^L P^2(i, j, d, \theta) \quad (4)$$

용어 정의4 엔트로피

$$H(d, \theta) = \sum_{i=0}^L \sum_{j=0}^L P(i, j, d, \theta) \log P(i, j, d, \theta) \quad (5)$$

용어 정의5 관성

$$I(d, \theta) = \sum_{i=0}^L \sum_{j=0}^L (i-j)^2 P(i, j, d, \theta) \quad (6)$$

용어 정의6 운동량

동영상 템포럴 텍스처 영상에서 움직임을 나타낸 픽셀의 집합을 M_{amount} 라 하고 해상도를 고려하여 정규화한 값을 M_{ratio_amount} 라 하면 운동량은 다음과 같이 정의된다.

$$M_{ratio_amount} = \frac{n(M_{amount})}{i * i} \quad (7)$$

여기서 $i * i$ 는 해상도

용어 정의7 운동 중심

동영상 템포럴 텍스처 영상에서 움직임을 나타낸 픽셀들의 집합에서 직접 구한 운동 중심을 M_{raw_center} 라 하고 x 좌표를 $M_{raw_center}(x)$, y 좌표를 $M_{raw_center}(y)$, 정규화한 운동 중심을 M_{ratio_center} 라 하면 정규화한 운동 중심은 다음과 같은 두 원소를 갖는 튜플로 정의한다.

$$M_{ratio_center}(x) = \frac{M_{raw_center}(x)}{i} \quad (8)$$

$$M_{ratio_center}(y) = \frac{M_{raw_center}(y)}{i} \quad (9)$$

여기서 $i * i$ 는 해상도

에너지는 명암도의 균일성을 재는 척도가 된다. 즉, 움직임이 균일하게 일어나는지를 표현하게 된다. $z = |i - j|$ 일 때 거리 d 만큼 떨어지고 명암도 차이가 z 인 픽셀쌍의 수를 $P_d(z)$ 라 하면 에너지는 z 의 변화에 대하여 거의 같은 경우 에너지의 값이 최소가 되고 한 속에 집중할 때 큰 값을 갖게 된다. 관성의 경우에는 명암도 차이가 크게 나는 픽셀의 수가 많으면 큰 값을 갖게 된다.

4.1 템포럴 텍스처 특성 벡터 생성

템포럴 텍스처의 특성을 나타내는 특징 벡터를 생성하는 기법으로는 두 가지 접근 방법을 사용한다. 첫 번째 방법은 앞에서 정의한 에너지, 엔트로피, 관성, 운동량, 운동 중심을 한 패턴으로 만들고 두 번째 접근 방법은 오브젝트 분석을

하여 기하학적인 특성을 저장하는 것이다. 그러나 템포럴 텍스처는 텍스처의 특성상 기하학적인 특성을 보이는 경우가 드물고 기하학적인 특성으로 동영상의 특성 벡터를 생성할 경우에는 움직임의 기하학적인 특성보다는 색이나 배경 특징, 혹은 템플릿을 이용한 움직임 오브젝트 분석 방법이 더 효율적이므로 본 연구에서는 기하학적인 특성은 고려하지 않았다.

본 논문에서 대상으로 하는 동영상은 두 가지의 가정을 갖고 있다. 첫째, 대상으로 하는 이미지 열은 그 내용이 하나의 이야기이다. 즉, 동영상에서 경계 샷(shot)을 구해야 하는 문제는 포함되지 않는다. 두 번째 가정은 움직이는 오브젝트가 한 개라는 것이다.

에너지, 엔트로피, 관성 계산시에는 $P_d(z)$ 는 z 에 대한 히스토그램이라고 볼 수 있으므로 히스토그램을 구하여 이용하면 효율적이다. 이 계산의 결과로 산출되는 값은 해상도에 영향을 많이 받으므로 해상도에 대한 비율로 표현하여야 한다. 운동량은 M 요소로부터 만들어진 템포럴 텍스처는 움직임에 참가한 픽셀만을 표현하므로 이러한 픽셀의 수를 세면 그 동영상에서의 운동량을 표현할 수 있다. 그러나 픽셀의 수로만 표현하면 이 경우에도 해상도에 민감하게 되므로 운동 비율로 표현하는 것이 바람직하다. 또한 운동 중심도 정규화하여 저장하여야 한다. 예를 들어 (100,100)이 운동 중심이면 해상도 128*128인 경우와 256*256 인 경우의 위치가 다르므로 식(8)(9)을 이용하여 정규화하였다. 표 3에 실험 결과의 일부를 보였다.

4.2 템포럴 텍스처의 운동 중심 보정

동영상에서 움직임이 여러 부분에서 나타나는 경우 여러 움직임 중 어느 한 움직임이 최소 행렬처럼 나타나는 경우에는 운동 중심에 왜곡이 발생하게 된다. 그림 11에서 화자가 주로 입을 움직여 말하는 동영상의 경우의 움직임을 보이는 영역을 나타냈다.



(a) 샘플링된 첫 프레임 (b) 샘플링된 마지막 프레임
(c) 생성된 템포럴 텍스처

그림 11 입을 주로 움직이는 동영상의 예
Fig. 11 Video example for speaking

이 템포럴 텍스처에서 구해진 운동 중심은 해상도가 256*256일 때 (206,142)이다. 그런데 이 위치는 입이 움직임의 중심이라고 인식하는 인간의 인지 과정과는 차이를 보인다. 그 이유는 이 동영상 속의 화자의 눈과 머리카락 등이 미세하게 움직이는 것도 템포럴 텍스처 안에 저장되어 있으며 이러한 움직임은 최소 행렬의 형태로 계산에 참여하게 되

어 결과적으로 움직임의 중심을 왜곡되게 하는 것이다. 따라서 이러한 경우 운동 중심을 보정하여야 하는데 다음과 같은 방법을 제안한다.

알고리즘 3 운동 중심 보정

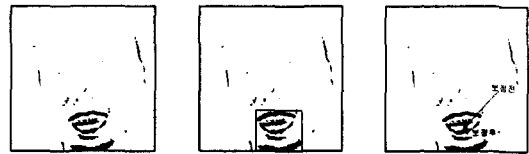
- 단계 1 : 알고리즘1에 의해 생성된 템포럴 텍스처를 이진 영상으로 변환한다.
- 단계 2 : 단계 1에서 생성된 이진 영상에서 임계치 이상인 정방향 사각형을 구한다.
- 단계 3 : 단계 2에서 구한 사각형안에 있는 움직임 정보만을 대상으로 운동 중심을 계산한다.

단계 2에서 구하는 정방향 사각형은 모든 픽셀에 대해 구한다면 계산 시간이 많이 소요될 것이다. 이 사각형을 구하는 효율적인 방법을 알고리즘 4에서 제시하였다.

알고리즘 4 운동 중심 윈도우 탐색

- 단계 1 : 템포럴 텍스처 영상을 $P * P$ 개의 부영역으로 나눈다. 여기서 각 부영역의 크기는 $p * p$ 이다.
- 단계 2 : 각각의 부영역에 대하여 운동량의 비율을 계산한다.
- 단계 3 : 각 부영역들에 대해 한쪽 에지의 크기가 p 의 배수인 정방향 윈도우를 만들어 각각의 윈도우 안의 운동량을 구한다. 이 때 운동량이 임계치 이상이면 멈춘다.
- 단계 4 : 윈도우가 결정되면 윈도우 안에 있는 실제 움직임을 보인 픽셀들을 대상으로 운동 중심을 다시 계산한다.

이 방법에 의한 보정 결과를 그림 12에 제시하였다.












(a) 운동 중심 왜곡을 보이는 경우 (b) 주 운동 영역 선택 (c) 운동 중심 보정 결과

그림 12 운동 중심 보정 예
Fig. 12 Example of motion centroid adjustment

템포럴 텍스처에서 그 텍스처를 특징지을 수 있는 벡터들을 계산해 낸다. 이 실험에서는 에너지, 엔트로피, 관성, 운동량, 운동 중심 등을 계산하였다. 그 결과를 표 3에 보였다.

표 3 동영상 템포럴 텍스처 예에 대한 특성 벡터 값
Table 3 feature vectors of temporal textures

템포럴 텍스처의 예 (각 그림 크기는 256*256)			특성 벡터값 (에너지, 엔트로피, 관성의 값은 d=90, z=10인 경우)	
샘플링된 첫 프레임	샘플링된 마지막 프레임	생성된 템포럴 텍스처		
			에너지	3.2735e+004
			엔트로피	8.8655
			관성	176.4645
			운동량 (비율)	9593(0.1464)
			운동중심 (비율)	(143.99,126.62) (0.56,0.49)
			에너지	2.9100e+004
			엔트로피	8.7521
			관성	204.0879
			운동량 (비율)	10547(0.1609)
			운동중심 (비율)	(150.49,140.35) (0.58,0.54)
			에너지	4.8891e+004
			엔트로피	9.5192
			관성	4.6936
			운동량 (비율)	15266(0.2329)
			운동중심 (비율)	(152.57,154.79) (0.59,0.60)

5. 오브젝트 운동 궤적 추출

템포럴 텍스처의 특성은 대상이 된 동영상의 움직임 정보만을 한 장의 그림에 담을 수 있다는 것이다. 앞 장에서는 이러한 템포럴 텍스처를 생성하여 이 텍스처에 공간 텍스처 특성을 적용하여 그 성능이 입증된 공간적 명암도 의존성 특성 중 에너지, 엔트로피, 관성 등을 계산하였으며 운동량과 운동 중심을 계산하였다. 또한 템포럴 텍스처에서는 오브젝트 운동 궤적을 추출할 수 있다. 이 운동 궤적은 동영상 검색 시스템에서 동영상 안에 있는 오브젝트가 움직이는 궤적에 관한 정보이므로 동영상을 검색할 때 필요한 정보이다.

5.1 운동 궤적의 수학적 표현

운동 궤적에 관한 연구는 기존의 연구에서 특히 궤적 연구(trajjectory)와 유사하다. 궤적 연구는 주로 두 가지 상이한 접근 방법으로 진행되어 왔다. 첫 번째 접근 방법은 궤적이란 오브젝트의 중심이 이동한 자취로 정의하며 정형적으로 기술하자면, 길이 n 인 궤적의 k 번째 프레임의 주 오브젝트의 중심을 (x_k, y_k) 라 하면 궤적은 튜플 $\{(x_k, y_k); k=1...n\}$ 의 집합이다. 이 함수는 해당하는 프레임들의 쌍에 대해 움직임 벡터 필드를 분할해 내고 트래킹을 통해 프레임들 사이

의 해당 오브젝트를 결정하여 구해진다[9]. 두 번째 접근 방법은 움직이는 궤적을 추출하기 위해 연속된 프레임에서 오브젝트가 분할되었다 가정하고 그 오브젝트의 centroid를 추적하는 다항식을 만들어 오브젝트의 궤적을 모델링하였다[10]. 이 연구에서는 n 이 근사도(approximation order)이고 계수의 개수는 $n+1$ 개이면 궤적을 표현하는 다항식은 다음과 같은 식과 같이 구해진다.

$$y = a_0 + a_1x + a_2x^2 + \dots + a_{n-1}x^{n-1} + a_nx^n$$

오브젝트 운동 궤적에서 추출해야 하는 정보는 오브젝트 운동 궤적의 위치, 오브젝트 운동 궤적의 크기, 오브젝트 운동 궤적의 모양 등 세 가지로 요약할 수 있다.

용어 정의 8 오브젝트 운동 궤적

이미지 열, 즉 동영상을 두 공간 차원 x, y 와 하나의 시간 차원 t 로 구성된 시공간입체(spatiotemporal solid)로 고려해 보면 움직임은 입체(image solid)에 있는 완만한 곡선을 따라 정보가 집중되게 된다. 이러한 곡선을 오브젝트 운동 궤적이라 한다.

5.2 오브젝트 운동 궤적의 계산 과정 구현

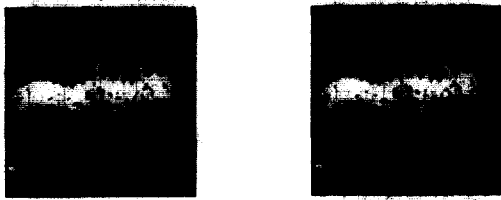
오브젝트 운동 궤적을 계산하기 이전에 우선 앞 장에서 생성해낸 템포럴 텍스처의 본질에 대하여 고찰할 필요가 있다. 템포럴 텍스처는 웨이브릿으로 변환된 압축 영상에서 움직임 정보를 포함하고 있는 M 요소를 병합하여 생성하였다. 따라서 템포럴 텍스처의 특성은 다음과 같다.

- 배경과 같은 불변의 정보는 삭제되었다
- 오브젝트의 특성은 알려져 있지 않다
- 움직임의 순서는 알 수 없다.
- M 요소의 계조값은 움직임의 순서, 크기, 특성 등에 무관하다.

이러한 특성에 의하면 결론적으로 템포럴 텍스처에서 추출되는 오브젝트 운동 궤적은 주기성을 가진 반복 운동은 만약 같은 궤적을 이동하는 움직임이라면 반복 운동인지 알 수 없다는 것이다.

5.2.1 참조 블록 크기 선택

오브젝트 운동 궤적을 추출하기 위해서는 용어 정의에서 언급한 3차원 입체에 있는 완만한 곡선을 탐색해야 한다. 이러한 곡선을 탐색하기 위해 모든 픽셀을 검사하는 것은 시간과 계산 과정의 낭비이다. 따라서 앞장에서 운동량과 운동 중심을 구할 때 사용한 것처럼 템포럴 텍스처 영상을 중복되지 않게 분할하여 탐색하는 것이 효과적이다. 그림 13은 본 연구에서 정규화의 단위로 사용하고 있는 256*256 크기를 부영역 크기 16*16과 32*32로 분할한 경우이다.



(a) 부영역 크기 16*16인 경우 (b) 부영역 크기 32*32인 경우
그림 13 참조 블록 크기

Fig. 13 Different examples of reference block size

부영역 크기 32*32의 경우는 참조 블록이 8*8개로 구성된다. 이 경우 그림 14과 같은 표기를 사용한다면 그림 13(b)의 경우는 ((4,1),(4,2),(4,3),(4,4),(4,5),(4,6),(4,7))이 된다.

(1,1)	(1,2)	(1,3)	(1,4)	(1,5)	(1,6)	(1,7)	(1,8)
(2,1)	(2,2)	(2,3)	(2,4)	(2,5)	(2,6)	(2,7)	(2,8)
(3,1)	(3,2)	(3,3)	(3,4)	(3,5)	(3,6)	(3,7)	(3,8)
(4,1)	(4,2)	(4,3)	(4,4)	(4,5)	(4,6)	(4,7)	(4,8)
(5,1)	(5,2)	(5,3)	(5,4)	(5,5)	(5,6)	(5,7)	(5,8)
(6,1)	(6,2)	(6,3)	(6,4)	(6,5)	(6,6)	(6,7)	(6,8)
(7,1)	(7,2)	(7,3)	(7,4)	(7,5)	(7,6)	(7,7)	(7,8)
(8,1)	(8,2)	(8,3)	(8,4)	(8,5)	(8,6)	(8,7)	(8,8)

그림 14 참조 블록 위치 표기
Fig. 14 Location notation of reference blocks

5.2.2 오브젝트 운동 궤적 제어점 선택

오브젝트 운동 궤적을 이루는 블록을 선택하기 위해 앞에서 사용한 움직임 정보의 크기, 즉, 운동량을 계산한다. 해당 블록의 운동량이 임계치 이상이면 움직임에 참가한 오브젝트의 일부라고 판단한다. 그리고 나서 해당 블록의 레이블을 오브젝트 운동 궤적 집합의 원소로 삽입한다. 오브젝트 운동 궤적을 집합으로 표시하는 이유는 기존의 연구에서 사용된 튜플 기법이나 다항식의 계수를 이용한 표기는 움직임의 순서에 근거한 표기법이다. 그러나 템포럴 텍스처를 이용한 오브젝트 운동 궤적은 움직임의 시간적인 인과 관계를 알 수 없기 때문에 순서를 고려하지 않는 표기로서 집합 원소 표기를 선택하였다.

5.2.3. 동일한 오브젝트 운동 궤적 판단

대상으로 삼은 동영상에 한 개의 오브젝트를 가지고 있다고 정의하였으므로 움직임 정보는 한 개의 오브젝트의 움직임에 관한 정보이다. 템포럴 텍스처는 동영상을 일정한 시간 간격으로 샘플링한 프레임에 관한 정보이므로 해당 오브젝트의 이동 속도에 따라 움직임 정보가 연속적으로 표현될지 혹은 고립된 정보로 표현될 지가 결정된다. 그러나 오브젝트의 시작 지점과 종료 지점이 같은 동영상에 대해 속도가 다르고 해도 운동의 방향이 변하는 것은 아니기 때문에 속도에 관계없이 참조 곡선은 동일한 것을 공유하게 된다. 또한 시

작점과 종료점이 같은 경우에도 운동 방향이 도중에 변하였다가 다시 돌아오는 경우에는 다른 오브젝트 운동 궤적을 갖게 될 것이다. 이러한 상황을 표현하기 위해 그래프 표기를 사용하였다. 그림 13(b)의 경우 움직임 곡선을 그래프로 표현하면 그림 15와 같다. 각각의 참조 블록을 정점(vertex)으로 하고 인접 블록을 에지(edge)로 연결하였다.

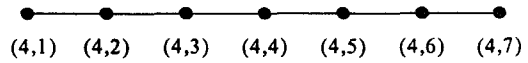


그림 15 그림 13(b)의 오브젝트 운동 궤적 제어점 그래프
Fig. 15 Trajectory control point graph of Fig. 13(b)

동일한 동영상에서 샘플링 간격을 두 배로 늘이면 만약 오브젝트가 등속으로 이동하는 물체였다면 (4,1), (4,3), (4,5), (4,7)만 샘플링되어 템포럴 텍스처를 구성하게 될 것이다.

이 경우 정점 (4,1)과 정점 (4,7)을 직선으로 연결하는 오브젝트 운동 궤적이 생기는 때 이것은 그림 15의 오브젝트 운동 궤적과 동일하다. 따라서 정점 (4,1)과 정점 (4,7)을 연결하는 직선을 오브젝트 운동 궤적으로 갖게 되는 경우는 다음과 같은 비 방향성 그래프로 표현할 수 있다.

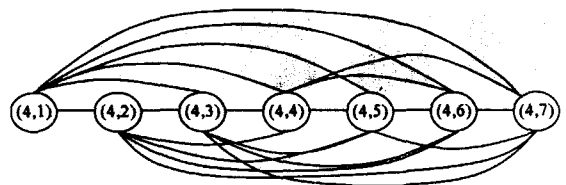


그림 16 동일한 오브젝트 운동 궤적을 갖는 그래프
Fig. 16 Graph notation for sharing the same trajectories

이 그래프에 의하면 시작 위치 (4,1)에서 시작하여 종료 위치 (4,7)에 도착할 때까지 반복 운동과 잠시 멈춤 등을 포함한 모든 운동은 동일한 오브젝트 운동 궤적을 공유하게 된다. 그러나 정점 (4,1)에서 시작하여 수평 방향 이외의 방향으로 방향을 바꾸었다가 다시 돌아와 정점 (4,7)에 도착하는 경우는 다른 운동 궤적을 갖게 된다.

6. 결론

움직임 정보는 동영상에서만 표현될 수 있는 중요한 특징이다. 본 논문에서는 동영상에서의 움직임 정보를 모델링하여 쉽게 동영상 검색 시스템에서 사용할 수 있도록 연구하였다. 제한한 템포럴 텍스처를 이용한 동영상 검색을 위한 움직임 모델링 기법은 다음과 같은 결과를 얻었다.

- 템포럴 텍스처 생성
- 텍스처의 특성 벡터 선택
- 템포럴 텍스처의 이동 궤적 추출
- 동영상 검색 시스템의 주석으로 이용

이렇게 생성된 템포럴 텍스처에서의 움직임 특성은 계산하기 간단하며 개념도 단순하여 동영상 검색 시스템에서의 주석으로 사용되어 인덱싱에 사용될 수 있다. 압축된 동영상에 직접 사용될 수 있으므로 실시간 검색에 사용될 수도 있을 것이다. 현재 사용되는 사람이 직접 수행하는 동영상의 의미 기술 기법은 점점 늘어나는 동영상 데이터의 양에 비추어 보면 비현실적이며 그 내용 또한 지나치게 주관적이므로 템포럴 텍스처와 같은 정량화된 움직임 표현 척도가 사용되어야 할 것이다.

본 논문에서 도출한 결과를 중심으로 다음과 같은 향후 연구 과제들을 제시한다.

- 텍스처 특성 벡터 선택
- 주기성을 가진 운동을 표현하는 표기법 개발
- 움직임 순서를 표현할 수 있는 칼라링 개발
- 궤적 그래프를 주석으로 이용

감사의 글

이 논문은 1998년 한국학술진흥재단의 학술 연구비에 의하여 지원되었음

참 고 문 헌

[1] Ramprasad B. Polana, "Temporal Texture and Activity Recognition", University of Rochester, Ph.D Thesis, 1994.

[2] Marcin Olof Szummer, "Temporal Texture Modeling", MIT Press, MS Thesis, 1995.
 [3] Fang Liu, "Modeling Spatial and Temporal Textures", MIT Press, Ph.D Thesis, 1997.
 [4] 최정구, "웨이브릿과 제로트리 구조를 이용한 개선된 정지 영상 압축 기법," 석사학위논문, 이화여자대학교, 1999.
 [5] 김도년, 조동섭, "동영상 검색을 위한 템포럴 텍스처의 특성 벡터 생성 알고리즘," 대한전기학회 추계 학술대회 논문집, 2000.
 [6] 장도임, 김도년, 강민숙, 조동섭, "Temporal 웨이브릿을 이용한 물체 추적 알고리즘," 대한 전기학회 하계 학술대회 논문집, vol.G. pp.2843-2845, 1999
 [7] Richard W. Connors, Charles A. Harlow, "A Theoretical Comparison of Texture Algorithms," IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. PAMI-2, No.3, pp.204-222, May, 1980.
 [8] 김도년, "템포럴 웨이브릿 기반 동영상 템포럴 텍스처의 특성 연구," 박사학위논문, 이화여자대학교, 2001
 [9] Emile Sahouria, "Video indexing based on object motion," Master's thesis, dept. of electrical engineering and computer science, Univ. of California at Berkeley, May 1997.
 [10] Kyuwon Lee, Woong-Shik You and Jinwoong Kim, "Video retrieval based on the object's motion trajectory," Proc. of visual communications and image processing 2000, pp.114-124, Apr. 2000.

저 자 소 개



김도년 (金度年)
 1989년 이화여자대 컴퓨터학과 졸업. 1991년 이화여자대 대학원 컴퓨터학과 졸업 (석사). 2001년 이화여자대 과학기술대학원 컴퓨터학과 졸업 (공학). 1992년 연세대의과대학 의용공학과 조교. 2001년- 현재 LG 전자기술원 선임연구원

Tel : 02-3277-2309, Fax : 02-3277-2306
 E-mail : dnkim@LG-Elite.com



조동섭 (趙東燮)
 1979년 서울대 공과대학 전기공학과 졸업. 1981년 서울대 대학원 전기공학과 졸업 (석사). 1986년 서울대 공과대학 컴퓨터학과 졸업 (공학). 1985년- 현재 이화여자대 컴퓨터학과 교수. 1992년 - 1995년 주전산기산학 협의회 부회장. 1993년 - 현재 대검찰청 과학수사자문위원회 자문위원. 1996년 University of California, Irvine 연구교수.

Tel : 02-3277-2314, Fax : 02-3277-2306
 E-mail : dscho@mm.ewha.ac.kr