

논문 2004-41C1-2-12

비디오 영상에서 시공간적 문자영역 제거방법

(Spatiotemporal Removal of Text in Image Sequences)

이 창 우*, 강 현*, 정 기 철**, 김 항 준*

(Chang-Woo Lee, Hyun-Kang, Kee-chul Jung, and Hang-Joon Kim)

요 약

많은 시각적 정보를 포함한 비디오 데이터들의 자동화된 처리 기술 중, 비디오 데이터들의 시청각적인 정보를 보강시키고, 부가적인 정보를 첨가하기 위한 일환으로 자막을 삽입하는 경우가 많다. 이러한 자막은 때로 영상자료의 재사용성(reusability)을 저해하고, 원 영상을 훼손하는 경우가 발생한다. 본 논문에서는 영상의 재사용성을 높이고 원 영상 복원을 위해 Support Vector Machines(SVM)과 시공간적 영상복원 방법(spatiotemporal restoration)을 이용한 비디오 영상에서의 자동 문자 검출과 제거 방법을 제안한다. 연속적인 두 프레임 이상의 영상을 입력 받아, 현재 프레임 영상에서 SVM을 이용하여 문자 영역을 검출한 다음, 검출된 문자 영역을 제거하고, 문자 영역에 의해 가려졌던 원 영상을 복원하기 위한 두 단계- 시간적 복원(temporal restoration)과 공간적 복원(spatial restoration)접근방법을 제안한다. 제안된 복원 방법은 글자 모션(text motion) 정보와 두 영상의 배경 차이(background difference)를 이용하여 영상을 그 특징에 따라 분류하고, 각 영상의 특징에 맞는 복원 방법을 적용한다. 제안된 방법은 다양한 종류의 영상에서 문자뿐만 아니라 관심의 대상이 되는 객체의 자동 검출 및 복원 등 다양한 응용분야를 포함한다.

Abstract

Most multimedia data contain text to emphasize the meaning of the data, to present additional explanations about the situation, or to translate different languages. But, the text makes it difficult to reuse the images, and distorts not only the original images but also their meanings. Accordingly, this paper proposes a support vector machines (SVMs) and spatiotemporal restoration-based approach for automatic text detection and removal in video sequences. Given two consecutive frames, first, text regions in the current frame are detected by an SVM-based texture classifier. Second, two stages are performed for the restoration of the regions occluded by the detected text regions: temporal restoration in consecutive frames and spatial restoration in the current frame. Utilizing text motion and background difference, an input video sequence is classified and a different temporal restoration scheme is applied to the sequence. Such a combination of temporal restoration and spatial restoration shows great potential for automatic detection and removal of objects of interest in various kinds of video sequences, and is applicable to many applications such as translation of captions and replacement of indirect advertisements in videos.

Keywords: Text detection, Text removal, Motion estimation, Spatiotemporal restoration, SVM

* 정희원, 경북대학교 컴퓨터공학과
(Dept. of Computer Eng. at Kyungpook National Univ.)

** 정희원, 숭실대학교 미디어학부 - 교신저자
(College of Information Science at Soongsil Univ.)

※ 이 논문은 2002년도 학술진흥재단의 지원에 의하여 연구되었음. (KRF-2002-037-D00008)

접수일자: 2003년9월24일, 수정완료일: 2004년3월2일

I. 서 론

최근 영화나 TV 프로그램의 시청 중에 특정 상표의 로고나, 옷 상표, 가구나 전자제품 등에 부착된 간접광고를 쉽게 접할 수 있다. 현재까지 이러한 간접 광고는 모든 촬영을 마친 후, 손으로 삭제하거나 밴드 등을 이용하여 사전에 가리고 촬영하는 경우가 대부분이었다. 이러한 전통적인 방법과는 달리, 자동으로 영상에서 이러한 불필요한 물체(예를 들면, 영상에서 얼룩이나 흠집, 스크래치(scratch), 잡음, 그리고 글자 등)를 검출하고 제거하는 몇몇 방법이 제안되었다.

비디오 영상에서 글자의 검출 및 제거 관련 연구로써, Kokaram et al.은 연속 영상에서 스크래치나 흠집의 검출과 제거에 관한 연구 결과를 발표하였다^[1]. 일반적으로 글자는 이러한 스크래치보다 더 큰 영역을 차지하고 있기 때문에 Kokarametal.의 방법을 자동 문자 검출 및 제거에 적용하기는 부적절하다. Bertalmio et al.은 자동 영상 복원을 위한 편미분 방정식(partial differential equation)을 이용한 인페인팅(inpainting) 방법을 제안하였다^[2]. 이 방법은 관심의 대상이 되는 영역을 미리 사용자가 정의해야 한다는 단점이 있으나, 영상에서 미리 정의된 영역이나, 임의로 삽입한 글자 영역을 사용자가 지정하여 그 지정된 영역에 의해 가려진 영역을 원 영상에 가깝게 복원하는 장점이 있다. Chan과 Shen은 필름이나, 사진 영상에서 흠집을 제거하기 위해 curvature driven diffusions을 이용한 영상복원 방법을 제안하였다^[3]. 또한 Wei와 Levoy는 텍스처 합성을 위하여 트리 구조를 가진 벡터 양자화 방법을 사용하는 영상복원 방법을 제안하였다^[4]. 이러한 방법들은 모두 복원하려는 영역 주변의 텍스처가 뚜렷해야 좋은 결과를 얻을 수 있다.

Irani와 Peleg은 영상 향상(image enhancement)과 겹친 영역복원을 위하여 연속 영상으로부터 추출한 모션정보를 이용한 복원 방법을 발표하였고^[5], Chun과 Bae는 자막의 교환을 위한 방법으로 자막에 의해 가려진 원 영상을 복원하기 위해 모션 보상 복원(motion-compensated recovery) 방법을 제안하였다^[6]. 이러한 방법들은 가려진 배경영역의 복원 결과는 주목할 만 하지만, 2장 이상의 연속 프레임들을 사용함에 따라 모션정보를 추출하기 위한 계산량이 많다. 또한 이러한 방법을 비디오 영상에서 문자 영역의 자동 검출 및 복원에 적용한다면, 움직이지 않고 고정된 문자영역에 의해 가려진 배경 영역을 복원할 수 없다.

본 논문에서는 비디오 영상에서의 자동 문자 검출과 제거 방법을 제안한다. 이를 위하여 연속 영상에서 문자 영역을 검출하는 기술과 검출된 문자 영역의 제거와 동시에 문자 영역에 의해 겹쳐진 배경영역을 복원하는 기술이 필요하다. 제안된 방법에서는 입력영상에서 문자 검출을 위해 SVM을 사용한다. 그 이유는 SVM이 기존의 분류기들 보다 일반화의 능력이 우수하다고 알려져 있고^[23-24], 실험에 사용되는 실험 데이터들의 다양성으로 인해, 일반적인 분류기들 보다 그 분류 능력이 우수한 SVM을 사용한다. 배경영역의 복원을 위하여 연속 입력 영상으로부터 추출된 2차원 모션정보를 사용하여 시간적 복원을 수행한 후, 시간적 복원으로 배경영역이 복원되지 않은 나머지 영역에 대하여 공간적 복원을 수행한다. 이와 같이 시간적 영상복원과 공간적 영상복원을 결합한 시공간적 영상복원 방법을 사용하는 이유는 다음과 같다.

- (1) MPEG과 같은 압축된 영상의 경우 문자 영역들이 심하게 훼손되어 있고, 글자영역과 배경영역과의 경계가 불분명하게 손상된 경우에는 문자영역에 의해 겹쳐진 영역의 복원을 위해 한 장의 영상으로부터 공간적 영상복원만으로 원 영상의 복원이 어렵다.
- (2) 배경영역의 복원을 위해 많은 연속 영상을 이용할 때, 매 프레임마다 모션정보를 추출해야 하는 부담이 존재하고, 또한 이에 따른 계산의 복잡도가 한 장의 영상을 이용하여 복원할 때보다 비교적 높다.
- (3) 만일 영상 내의 글자 영역이 움직이지 않을 경우에는 모션정보의 추출이 불가능하기 때문에 시간적 영상복원 방법을 사용할 수 없다.

[그림 1]에서는 시간적 영상복원 방법과 공간적 영상복원 방법을 결합하여 자동 문자 영역 검출 및 제거하는 방법의 개괄을 보여준다. 시퀀스 분류 단계에서는 글자의 모션정보와 배경의 모션정보를 활용하여 입력 비디오 시퀀스를 미리 정의된 네 개의 영상 Type 중에 하나로 분류한다. 이렇게 함으로써 배경 영역의 모션정보 추출 유무를 결정하고, 불필요한 배경영역의 모션정보를 계산하는 시간을 절약함과 동시에 잘못된 모션정보에 의한 영상복원의 오류를 방지할 수 있다.

본 논문의 순서는 다음과 같다. II장에서 입력영상에서 문자에 포함된 픽셀들의 텍스처 정보를 이용한 문자 검출 방법과 후처리 방법을 기술한다. III장에서는 검출된 문자영역에 의해 겹쳐진 배경영역의 복원을 복원하는 방법을 설명하고, IV장에서는 제안된 방법으로 복원한

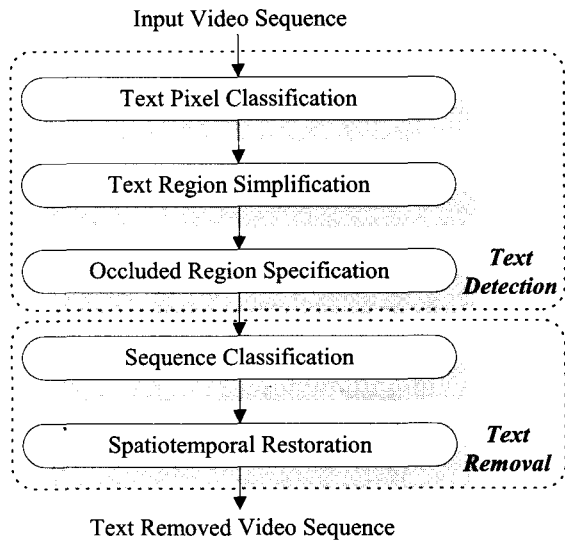


그림 1. 시공간적 영상복원 방법의 흐름도
Fig. 1. Flow of the combined technique for spatiotemporal restoration.

실험 결과에 대하여 논하고, V장에서는 임의로 구성된 시퀀스를 대상으로 제안된 방법을 비교, 분석, 평가한다. 마지막으로 VI장에서 결론을 맺고, 향후과제에 관하여 언급한다.

II. 문자 검출

문자영역을 검출하는 기존의 연구들은 크게 두 가지 범주로 분류할 수 있다. 첫째가 연결성분석 방법 (connected component-based method)에 의한 문자 검출 방법^[7-9]과 텍스처 기반의 문자 검출 방법^[10-17]으로 구분할 수 있다. 연결성분 기반의 방법들은 한 입력영상에 포함된 각각의 연결 성분들이 독립적으로 명세될 때까지, 연속적으로 작은 연결 성분들을 하나의 큰 연결성분으로 결합하는 방법이다. 이렇게 만들어진 연결 성분들은 각각 하나의 글자에 포함된 성분들의 기하학적인 정렬관계를 분석한다. 반면에 텍스처 기반의 방법들은 문자 영역을 특정 텍스처가 있는 물체(영역)로 간주하고, 입력영상에서 텍스처 성질(공간적인 변화(spatial variance), 공간적 주파수 등)들을 찾아내기 위해 Gabor filters나 Wavelet, FFT와 같은 방법들을 적용한다. 이렇게 입력영상으로부터 문자영역을 검출하기 위하여 텍스처적인 정보를 활용하는 것은 문자들의 폰트의 변형이나, 문자의 크기 그리고 문자를 표현하고 있는 색상 등에 아주 민감하기 때문에 적용한 응용에 따

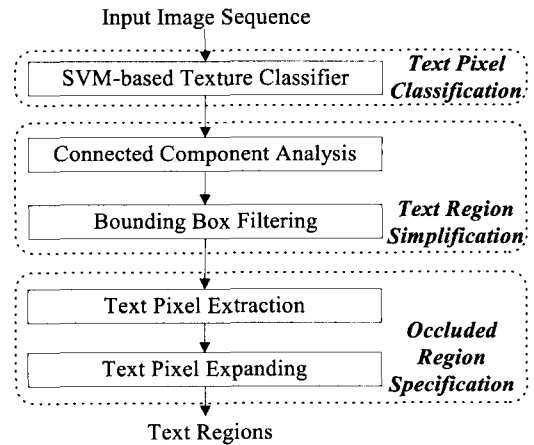


그림 2. SVM을 이용한 문자영역 정의의 흐름도
Fig. 2. Flow of text region specification using SVM.

라 특정 텍스처를 찾기 위한 텍스처 검출기를 만들기 어렵다. 본 논문에서는 비디오 영상에서 문자를 검출하기 위해 SVM을 이용한 자동화된 텍스처 분류기를 설계하고, 이를 입력영상에서 문자 검출기로 사용한다.

본 논문에서는 입력 영상 시퀀스로부터 문자 영역을 검출하고, 검출된 문자 영역이 가리고 있던 배경영역을 복원하는 것이 목적이다. 따라서 문자영역을 정의하는 단계가 선행되어야 한다. 이를 위해 [그림 2]와 같이 3단계의 접근방법을 사용한다. 첫째, 문자인 픽셀들과 문자가 아닌 픽셀들의 분류단계, 둘째, 문자영역을 조작을 쉽게하기 위한 문자영역의 단순화 단계 그리고, 셋째, 실제 복원할 배경영역을 결정하게 되는 문자영역 정의단계로 구성된다. 먼저 입력 영상 시퀀스의 현재프레임에서 SVM을 이용하여 문자인 픽셀과 문자가 아닌 픽셀로 분류한다. 분류된 이진 영상으로부터 문자 영역의 단순화를 위해 4방향 연결성분 조사과정을 거치고, 각 연결 성분이 미리 정의된 휴리스틱을 이용하여 글자가 아닌 연결성분들을 필터링한다. 마지막 단계에서 필터링된 영역을 잠정적인 문자영역으로 정의하고, 이 영역으로부터 실제 제거될 문자 영역을 결정한다. 이때, 잠정적인 문자영역들로부터 글자인 픽셀들의 색깔에 관한 단순한 가정으로 글자인 픽셀들만을 추출한다. 또한 모폴로지 연산(팽창연산)을 사용하여 주변의 노이즈의 제거와 영역내부의 구멍 부분을 보강한다. 본 장에서는 입력영상에서 문자를 검출하기 위한 방법을 간략하게 설명하고, 자세한 내용은 참고문헌[18-20]을 참조하기 바란다.

1. 텍스트처 분류기로서의 SVM

최근 SVM이 패턴 분류(pattern classification)와 비선형 함수의 추정(non-linear regression)을 위한 방법으로 제안되었다. 수식(1)과 같이 양성 샘플(positive sample, +1)이나 음성 샘플(negative sample, -1)로 레이블이 붙은 훈련 샘플들이 주어지면, SVM은 이 훈련 샘플을 이용하여 수식(2)의 제약조건 하에서 최적의 하이퍼평면(hyperplane)을 찾는 방법이다.

$$(x_i, y_i) \in \mathbb{R}^N \times \{\pm 1\}, i=1, \dots, l. \tag{1}$$

여기서 x_i 는 i 번째 입력 패턴이고, y_i 는 그 입력 패턴에 상응하는 기대 출력값이다.

$$y_i (\mathbf{w}^T \mathbf{x}_i + b) \geq 1, \text{ for } i=1, 2, \dots, N. \tag{2}$$

여기서 \mathbf{w} 는 가중치 벡터(weight vector)이고, b 는 편중치(bias)이다. 다시 말해, SVM은 $\Phi: \mathbb{R}^N \rightarrow F$ 의 매핑을 통해 입력 공간과 비선형으로 연관된 특징공간 F (feature space) 상의 선형 분류기이다. 원래 SVM은 선형 분류기이기 때문에, 비선형 문제에 적용하기 위하여 분류하고자 하는 데이터들을 비선형으로 F 공간으로 매핑하고, 특징공간에서 SVM을 이용한 선형 분류가 가능하게 된다. 특징공간 F 에서 데이터들이 선형으로 분류 가능하다는 것을 보장하기 위해 F 공간의 차원이 커야만 하고^[21], 특징공간 F 에서 SVM의 계산과 입력 데이터의 고차원의 매핑에 소요되는 실질적인 계산들은 커널 함수(kernel function)들의 활용으로 간접적이고 효과적으로 수행된다^[20]. 따라서 분류기는 특징공간 F 에서 모든 클래스 각각의 convex hull 사이의 거리인 마진(margin)을 최대화하면서 데이터 포인트들 사이의 가장 큰 부분을 정확하게 분리하는 기준이 되는 하이퍼 평면(canonical hyperplane)을 표현한다^[22]. 그리고 이 분류기는 입력 공간에서 아래 수식(3)과 같이 표현된다.

$$f(\mathbf{x}) = \sum_{i=1}^l y_i \alpha_i \Phi(\mathbf{x}_i) \cdot \Phi(\mathbf{x}) + b. \tag{3}$$

여기서 $\{\mathbf{x}_i\}_{i=1}^l$ 들은 최적 하이퍼 평면에서부터 가장 가까운 샘플들인 x_i 들을 표현하고 있고, 모든 α_i 가 0이 아닌 패턴들에 상응한다. 이 샘플들은 훈련 샘플의 부분 집합이고, Support vector(SV)라 부른다. 또한 α_i 와 b 는 이차 방정식 문제를 해결함으로써 결정되어지는 계수들이다^[22].

SVM의 매력은 저변에 깔려있는 통계적 학습이론(statistical learning theory)과의 밀접한 관계에 있다. 결과적으로 많은 응용분야에서 SVM는 신경망이나 radial basis function network들과 같은 전통적인 방법들 보다 더 좋은 일반화 능력을 보여왔다^[23-24].

계산상의 효율성을 위하여 매핑 Φ 은 흔히 수식(4)와 같이 정의된 커널함수 k 를 사용하여 묵시적으로 수행된다.

$$k(\mathbf{x}, \mathbf{y}) = \Phi(\mathbf{x}) \cdot \Phi(\mathbf{y}). \tag{4}$$

적절한 커널 k 를 선택함으로써 다양한 매핑(혹은 특징추출), Φ 은 간접적으로 유도된다^[22]. 특징추출을 위한 한 방법으로 입력 벡터 \mathbf{x} 의 각 원소인 x_i 들 사이의 p 차(order) 상관관계를 계산함으로써 구할 수 있다. 만약 \mathbf{x} 가 화소값으로 구성된 영상의 패턴이라 할 때, p 차 상관관계는 입력공간을 입력 화소들의 p -th order products 단항식으로 구성된 공간으로의 매핑과 같다. 이러한 특징들은 단순히 모든 상관관계를 단순한 계산에 의해서는 추출될 수 없다. 왜냐하면 $p < 2$ 일 때에도 요구되는 계산량이 상당히 많기 때문이다. 예를 들어 입력 패턴들이 N 차원이라면, 특징공간 F 의 차원은 $(N+p-1)!/p!(N-1)!$ 이다. 그러나 수식(5)와 같이 다항 커널의 도입으로 인하여, 특징공간에서의 작업들을 수행할 수 있고, p 차 다항식 커널 $k(\mathbf{x}, \mathbf{y}) = (\mathbf{x} \cdot \mathbf{y})^p$ 은 단항식 특징추출기에 의해 추출된 특징벡터의 내적과 같다^[22].

$$(\Phi_p(\mathbf{x}) \cdot \Phi_p(\mathbf{y})) = \sum_{i_1, \dots, i_p=1}^N x_{i_1} \dots x_{i_p} y_{i_1} \dots y_{i_p} = \left(\sum_{i=1}^N x_i y_i \right)^p = (\mathbf{x} \cdot \mathbf{y})^p. \tag{5}$$

텍스트 처리 작업을 수행할 때, SVM은 잘 알려진 다중 채널 필터링의 구현으로 볼 수 있다^[19]. SVM에서 다항식 커널의 동작은 한 채널에서 예지성분이나, 방향성과 같은 특징을 추출하는 채널의 동작과 근본적으로 같다. SVM에서 커널의 동작은 입력벡터 \mathbf{x} 와 SV(\mathbf{x}')의 내적을 계산하고, 비선형 매핑을 수행한 결과이다. 이 경우 SV들은 각각의 특징들을 추출하는 필터의 역할을 한다. SVM에서 커널들이 특정 주파수대나 방향성 등의 특징을 추출하기 위한 필터로서 설계되지 않았다고 할지라도 SV들은 여전히 분류 목적을 위한 중요한 정보들을 추출할 수 있을 뿐 아니라 분류의 척도(measure)이다^[19]. 또한, 이러한 SV들은 특정 텍스트 분류 문제에 대해 일반

화 오류(generalization error)의 기준을 충족하면서 자동적으로 그리고 최적으로 선택된다. 텍스처 분류를 위해 최적화된 SV들을 선택한 SVM은 주어진 텍스처 분류문제를 위한 최적의 특징 추출기인 동시에 실험오류(test error)를 최소화하는 기준을 만족하는 텍스처 분류기이다.

이러한 SVM의 성질들을 이용하면, 텍스처 분류를 위한 SVM을 만들 때 미리 텍스처 성질에 맞는 특징을 추출하고, 그 특징들을 SVM의 입력으로 사용하는 과정을 배제함과 동시에 특징 추출 과정을 배제할 수 있다는 장점이 있다. 즉, SVM은 고차원의 특징공간에서 텍스처 패턴의 일반화를 잘 수행할 수 있고, 입력차원을 처리하기 용이한 저차원으로 줄이는 부담을 줄일 수 있다. 또한, SVM은 SVM 구조의 내부에 특징 추출기를 내포하고 있는 형태로 입력 패턴의 분류를 위한 특징으로써 입력패턴의 비선형 매핑, $\Phi(x)$ 나 $\theta(x \cdot x^*)$ 을 사용할 수 있다.

[그림 3]과 같이 SVM을 이용한 텍스처 분류기는 전혀 다른 역할을 하는 세 개의 층으로 구성된다. 먼저 입력층은 SVM과 외부의 환경을 연결시켜주는 역할을 하는 소스노드(source node)들로 구성된다. 입력패턴 x 는 입력 영상의 $M \times M$ 윈도우 내부의 픽셀 값을 직접 사용한다. 계산상의 효율성과 분류 성능의 향상을 위해 한 윈도우 내부의 모든 픽셀을 사용하지 않고, [그림 3]의 좌측 하단에서 보는 것처럼 회색 영역의 픽셀 값만을 사용한다.

이러한 패턴은 훈련단계에서도 동일하게 적용된다. 따라서 SV들은 그림에서 회색영역의 픽셀들만을 포함한다.

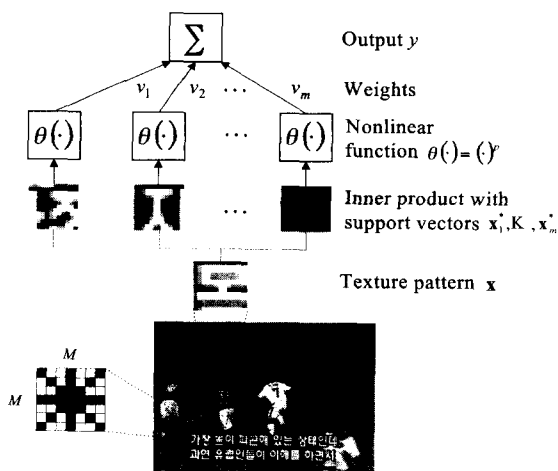


그림 3. SVM기반의 텍스처 분류기의 구조.
Fig. 3. Architecture of texture classifier.

다. 이렇게 함으로써 특징벡터의 크기도 M^2 에서 $4M-3$ 으로 줄일 수 있고, 또한 일반화의 성능을 높일 뿐 아니라 분류과정에서 계산 속도를 향상시킬 수 있다. 둘째로 은닉층은 SV들과 입력패턴 사이의 내적을 계산하고, 그 결과를 비선형함수에 적용시킨다. 은닉층의 크기인 m 은 훈련단계에서 결정된 SV의 숫자와 같다. 실제 테스트 단계에서는 입력 영상의 모든 픽셀을 $M \times M$ 윈도우의 중심으로 설정하여 전체영상을 탐색 하여, 각 픽셀이 문자 픽셀인지를 검사한다. 마지막으로 출력층에서는 입력패턴과 SV(x_i^*)들 사이의 내적을 계산하고, [그림 3]과 같이 계산된 각각의 결과에 가중치를 부과한 합($y_i \alpha_i$)의 형태로 출력을 계산한다. 출력의 부호, y 가 한 윈도우의 중심픽셀이 글자인지 여부를 표현한다. SVM의 훈련단계에서 +1은 문자 클래스로, -1은 비문자 클래스로 기대출력을 설정하였다. 즉, 만일 SVM의 출력값이 양수이면 그 픽셀은 글자로 분류된다.

SVM의 훈련단계에서 조절되어야 하는 파라미터는 다항식의 차수 p 와 입력윈도우의 크기 M 이다. VC 차원의 관점으로 볼 때, 텍스처 분류에 사용된 SVM의 정확도는 자동 분류 능력 제어에 있다. 따라서 분류 능력의 제어는 특징공간의 선택으로 사전에 명세된 일단의 함수들인 $\{f_a : a \in A\}$ 의 범위 내에 있다. 이것은 제한된 방법에서 p 와 M 의 결과와 동일한 일이다. 2000개의 문자 패턴들과 비문자 패턴들을 이용한 텍스처 분류의 경우처럼 훈련오류가 무시할 만할 때, VC 차원의 상한은 벡터 w 의 크기와 모든 데이터 포인트들을 포함하는 제일 작은 원의 반지름 r 의 제곱승에 비례하여 추정될 수 있다. r 과 w 는 특징공간 F 에 의존하기 때문에, VC 차원은 p 와 M 의 값에 의존한다. r^2 과 $\|w\|^2$ 은 특징공간 F 에서 직접 작업하지 않고, 커널의 사용에 의해 계산될 수 있다^[25]. 따라서 VC 차원의 추정된 상한은 최적의 파라미터를 획득하기 위해 최소화 되어야 하는 평가함수를 통해 구할 수 있다. 본 논문에서 사용된 SVM에서는 p 의 범위를 {1, 2, 3, 4, 5}, 그리고, M 의 범위를 {5, 7, 9, 11, 13, 15, 19, 21}로 한 40개 조합에 대한 실험 결과로 최적의 파라미터로 $p=3$, $M=13$ 으로 설정하였다.

2. 문자영역의 단순화

문자 영역 단순화 단계의 출력은 문자 영역들을 표현하는 사각형 박스들이다^[13]. 4 방향 연결성분(4-connected component) 분석을 통해 잠정적인 문자 영역을 추

출한 후, 각 영역에 포함된 문자픽셀의 비율을 통해 사각형의 경계를 조정한다. 예를 들어, 왼쪽 경계선은 박스 내부의 수직 프로젝션 프로파일을 조사한 후, 왼쪽으로부터 사각형 박스의 넓이의 1/10을 초과한 지점으로 새로운 왼쪽 경계면이 설정된다. 오른쪽, 위쪽, 아래쪽, 경계면들도 모두 같은 전략으로 수정된다. 모든 사각형 박스의 경계면들을 수정한 후, 아래의 세 가지 휴리스틱을 사용하여 잠정적인 문자 영역의 집합에서 문자 영역이 아닌 영역들을 제거하게 된다. 첫째, 한 문자영역 내에 포함된 문자 픽셀의 수가 그 박스 면적의 1/3 이상이어야 한다. 둘째, 한 문자 영역의 높이가 입력 영상의 높이의 1/30 보다 커야 한다. 셋째, 한 문자 영역의 넓이는 그 문자 영역의 높이의 절반 이상이어야 한다. 앞서 언급한 세 가지 휴리스틱에 위배되지 않은 사각형 영역들만이 문자 영역으로 검출한다. [그림 4]는 문자영역 검출의 중간 결과이다. [그림 4]의 (a)는 입력영상, (b)는 SVM으로 분류된 문자 픽셀이 흰색으로 표현되어 있는 출력 영상이며, (c)는 연결 성분들을 표현한다. 각 사각형 박스들은 잠정적인 문자 영역들이다. [그림 4]의 (d)는 문자 영역검출의 결과 영상이다. 여기서 각 영역들의 경계면들이 수정되고, 앞서 언급한 휴리스틱에 위배된 각 연결 성분들을 제거한 영상이다.

3. 문자 영역 추출

문자영역에서 문자 픽셀들만을 추출하기 위해 문자의 색상이 배경영역의 색상보다 밝다는 가정 하에 임계치 정책을 사용한다. 일반적으로 한 영상에서 배경과 문자를 구별하기 위해서 문자 주변에 문자 색깔과 대비되는 색깔로 문자를 두각(highlight)시키는 경우가 있다. 예를 들어, [그림 5]의 (a)는 삽입된 문자가 배경의 유사한 색깔들과 구별하기 위해서, 또는 그 문자를 강조하기 위해서 인위적으로 삽입된 검정색 경계를 보인다. 문자를 구성하고 있는 픽셀들의 경계부분이 인위적으로 삽입된 픽셀들에 의해 훼손되거나 영상의 압축과정 동안 이러한 픽셀들이 문자 영역을 훼손하는 경우가 있다. 이와 같은 경우에 문자영역을 공간적 복원방법만을 사용하면 신뢰할 만한 결과를 얻기가 어렵다. 왜냐하면 이와 같은 결과는 공간적 복원 방법이 복원할 영역의 주변영역의 픽셀 정보를 사용하기 때문이다. 따라서 제안된 방법의 공간적 복원 방법에서는 문자를 두각시키기 위해 삽입된 픽셀들과 문자 영역내부의 작은 구멍을 모두 포함하기 위해 팽창 모폴로지 연산(morphological operation

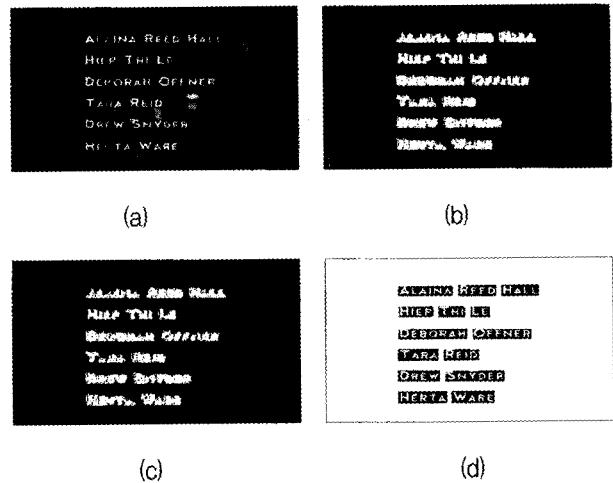


그림 4. 문자검출의 중간결과; (a) 720×400 크기의 입력 영상, (b) SVM의 출력, (c) 연결성분, (d) 문자영역
 Fig. 4. Intermediate results for detecting text regions: (a) input image sized 720×400, (b) output of the SVM filter, (c) 4-connected component, and (d) detected text.

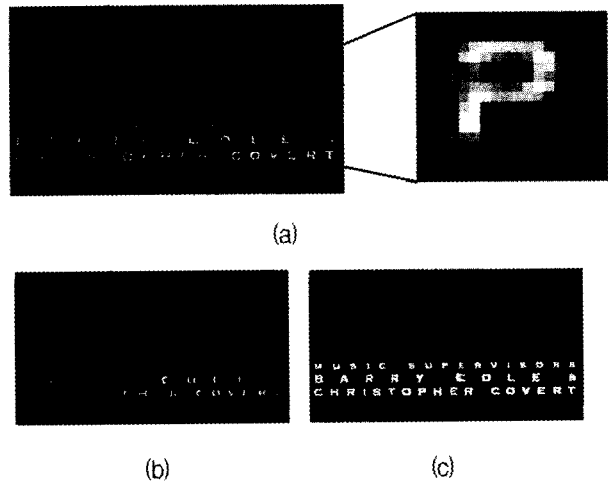


그림 5. 문자추출; (a) 인위적으로 삽입된 픽셀의 예, (b) 추출된 문자픽셀, (c) 확장된 문자 영역
 Fig. 5. Results of occluded region specification: (a) example of highlighting pixels, (b) extracted text pixels, and (c) result of expanding.

(dilation))을 사용하여 문자 픽셀 주변을 확장하여 확장된 문자픽셀을 구한다. [그림 5]의 (a)는 문자를 두각시키기 위해 삽입된 서너 개 정도의 픽셀들을 예시하고, (b)에서 흰색의 픽셀들은 문자를 두각시키기 위해 삽입된 픽셀들을 포함하지 않은 문자 픽셀들을 예시한다. [그림 5]의 (c)는 (b)에서 추출된 문자 픽셀들을 3×3 마스크를 사용한 팽창연산 후의 확장된 문자픽셀들을 보여 준다.

III. 시공간적 복원 방법을 이용한 문자 제거

본 장에서는 시공간적인 복원 방법에 관하여 설명한다. 제안된 방법은 시간적 복원과 공간적 복원의 결합된 형태로 문자영역을 제거하고, 그 배경영역을 복원하는 방법이다. [그림 6]에서 보는 바와 같이 시공간적 글자영역 제거 및 그 배경영역 복원 방법은 시퀀스의 분류와 분류된 시퀀스의 Type에 따라 각기 다른 복원 방법을 사용함으로써, 복원 결과의 정확도 및 계산시간의 단축 효과를 볼 수 있다.

먼저 정의된 문자영역의 모션을 추정하고, 추정된 모션에 따라 입력 시퀀스가 글자의 움직임이 있는지 여부를 판별한다. 또한 두 번째 단계로 배경영역의 움직임을 판별하기 위하여 적응적 임계치 방법^[27]을 사용하여 배경영역의 움직임 여부를 판별한다. 일반적으로 프레임 차이를 보고 움직임 여부를 판단할 경우, 영상의 압축이나, 영상의 획득 시 첨가되는 노이즈로 인하여 전역적인 차이를 보이기 쉽다. 따라서 이러한 노이즈에 의한 움직임을 제거하기 위하여 제안된 방법에서는 적응적 임계치 방법을 사용하여 배경영역의 움직임 여부를 판단한다. 분류된 시퀀스의 Type에 따라 선별적으로 배경모션의 추정 및 시간적 복원의 수행 여부를 판단하고, 각 Type에 따라 다른 시공간적 복원 알고리즘을 수행함으로써 복원의 정확도를 높일 수 있다. [그림 6]에서 M_T , M_B , $T1$, $T2$ 는 각각 문자 모션, 배경모션, 문자모션의 임계값, 배경모션의 임계값을 나타낸다.

1. 연속 영상의 분류

연속 영상에 포함된 대부분의 문자들은 몇 가지 특징이 있다. 예를 들어, 문자들은 때때로 움직임이 없이 고정된 위치에 존재하거나, 연속된 프레임에서 수직이나 수평으로 스크롤하는 경우가 있다. 그래서 이러한 특징을 시공간적 복원을 위해 사용한다면 계산상의 효율성을 높이며 더욱 정확한 모션 정보를 추출할 수 있다. 예를 들어, 배경영역이 움직이지 않고 고정되어 있는 상태의 문자라는 사실을 미리 알 수 있으면, 배경영역의 모션 정보를 구하는데 소요되는 시간이 더 이상 필요하지 않는 것이다. 또한 이와 같은 경우에 카메라나 영상에 포함된 물체에 의한 실제 움직임이 아닌 잡음에 의한 영상차이(frame difference)가 존재하는 경우에, 잘못된 배경영역의 모션정보를 구하는 일을 방지할 수 있고, 파생되는 잘못된 시간적 복원의 오류를 피할 수 있다. 이 단계에서

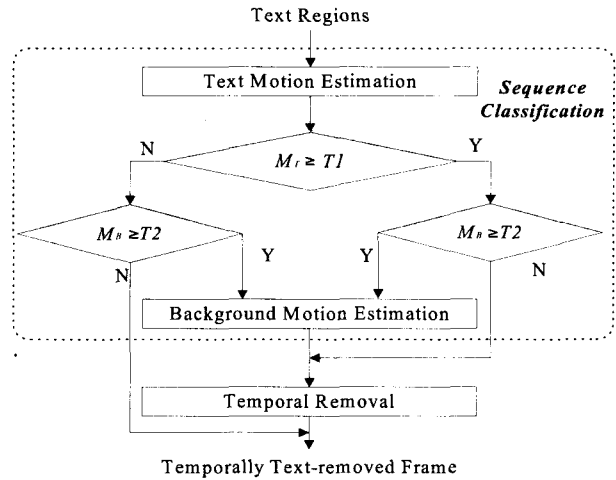


그림 6. 시공간적 글자영역 제거 및 배경영역 복원의 흐름도.
Fig. 6. Flow of spatiotemporal removal of text.

표 1. 연속 영상 분류 규칙.

Table 1. Four-type classification rules of a sequence for spatiotemporal restoration.

문자 모션	배경 차이	분류결과
$< T1$	$< T2$ 고정배경 상의 고정문자 시퀀스(Type 1 시퀀스)	
	$\geq T2$ 동적배경 상의 고정문자 시퀀스(Type 2 시퀀스)	
	$< T2$ 고정배경 상의 동적문자 시퀀스(Type 3 시퀀스)	
	$\geq T1$	$\geq T2$ 동적배경 상의 동적문자 시퀀스(Type 4 시퀀스)

는 [표 1]과 같이 문자의 움직임 정보(문자모션)와 배경 영상 차이를 이용하여 연속 영상을 분류한다.

추정된 문자모션이 임계치인 $T1$ (본 논문에서는 1로 설정하였다)보다 크거나 같으면 두 입력영상사이에서 문자가 움직였고 문자모션이 존재하기 때문에 이 시퀀스는 동적문자 영상(moving text sequence)으로 분류한다. 문자의 움직임 여부가 결정되고 난 후, 배경영역의 움직임을 판단하기 위해 연속된 두 프레임에서 배경영역의 차를 계산한다. 계산된 차이가 $T2$ 보다 작으면 입력 영상은 고정 배경영상으로 분류된다.

문자의 움직임과 배경의 움직임이 결정되면, 배경영역 차이가 있을 경우에만 블록모션 알고리즘(block motion algorithm, BMA)를 이용하여 배경모션(background motion)을 계산한다^[26]. 두 번째 임계값인 $T2$ 를 계산하기 위하여 모션의 검출을 위한 적응적 임계치 알고리즘(adaptive thresholding technique)을 사용한다^[27]. 적응적

임계치 방법은 두 연속된 영상에서 변화된 부분을 검출할 때 영상의 특징에 따라 임계값을 적응적으로 선택하는 방법으로 모션 검출을 위한 히스토그램 기반의 방법이다. 특히 이 방법은 영상 내에 포함된 물체의 움직임, 또는 카메라의 움직임으로 인한 영상차이가 아닌, 조명의 변화나 잡음에 의한 차이에 대한 민감성을 둔화시킨다. [그림 7]은 고정 배경 상의 움직이는 문자 영상열의 예를 보여준다. [그림 7]의 (a)와 (b)는 각각 현재 프레임과 이전 프레임을 나타내고, (c)는 두 영상의 차이를 그레이레벨로 표현한 영상이다. [그림 7]의 (d)는 적응적 임계치 알고리즘에 의해 생성된 임계값 12를 적용하여 구한 차이 영상이다. 즉, 흰색으로 표현된 픽셀들은 두 영상의 차이가 +12보다는 크고, 혹은 -12보다 작은 픽셀들이다.

2. 모션 추정

제안된 방법에서 문자모션을 추정하기 위하여 영역기반의 모션 모델을 사용하고, 배경모션을 추정하기 위하여 블록기반의 모션모델을 사용한다. 제안된 방법에서는 수식(6)과 같이 모션벡터를 추정하기 위하여 문자영역들이 선형으로만 움직인다고 가정하고 단순 이동 모델을 사용하여 모션(minimum absolute difference (MAD))을 추정한다^[26].

$$MAD(d_1, d_2) = \frac{1}{N_1 N_2} \sum_{(n_1, n_2) \in B} |I(n_1, n_2, t) - I(n_1 + d_1, n_2 + d_2, t - 1)| \quad (6)$$

수식(6)에서 B 는 후보 모션 벡터 (d_1, d_2) 를 계산하기 위한 $N_1 \times N_2$ 블록, t 와 $t-1$ 은 현재 프레임과 이전 프레임을 표현한다. 특히 문자 모션을 추정할 때는 B 는 문자영역을 나타내고, $N_1 N_2$ 는 한 문자 영역 내에 포함된 픽셀의 개수를 의미한다. 차이의 추정은 아래 수식(7)과 같다.

$$\arg \min_{(d_1, d_2)} MAD(d_1, d_2) \quad (7)$$

수식(7)에서 $-M_1 \leq d_1 \leq M_1$ 과 $-M_2 \leq d_2 \leq M_2$ 이다. M_1 과 M_2 는 각각 탐색 윈도우의 세로길기와 가로길기를 의미한다. 제안된 방법에서는 M_1 과 M_2 를 ± 16 으로 설정하고, 해당 탐색 영역을 모두 검색하여 모션 필드를 계산한다. 배경모션을 계산하는 단계에서는 어느 배경영역 블록이 한 픽셀이라도 문자 픽셀을 포함하면 배경영역으로

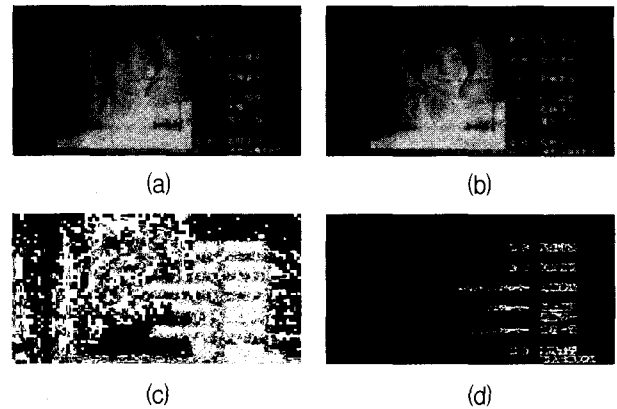


그림 7. 적응적 임계치 알고리즘의 수행결과; (a) 현재 프레임, (b) 이전 프레임, (c) 프레임 차이영상, (d) 적응적 임계치 알고리즘 적용 후의 프레임 차이 영상.

Fig. 7. Results of adaptive thresholding on the moving text on stationary background sequence; (a) current frame, (b) previous frame, (c) frame difference image, and (d) frame difference image after applying an adaptive thresholding technique.

간주하지 않고 모션추정을 하지 않는다. 대신에 수식(8)을 사용하여 겹쳐진 영역의 모션, MAD_o 을 계산하여 배경모션을 대체한다. 즉, MAD_o 는 문자 픽셀이 포함되어 있지 않은 인접한 주변 8 블록의 배경모션의 평균으로 문자 블록이 포함된 블록의 모션 벡터를 대체한다.

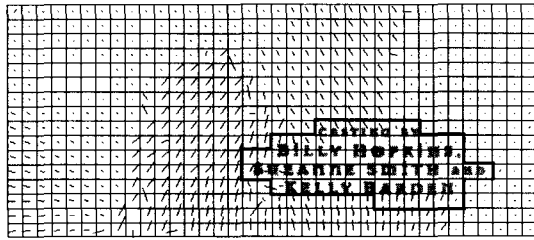
$$MAD_o = \frac{1}{n} \sum_{k \in TB} MAD_k(d_1^b, d_2^b) \quad (8)$$

수식(8)에서 k 는 인접 블록의 인덱스이고, TB 는 문자 블록을 의미한다. 그리고 n 은 문자가 포함된 블록이 아닌 인접한 블록의 개수이다. 만일 하나의 문자블록을 둘러싼 배경영역에 모션이 존재하지 않으면 겹쳐진 영역의 모션을 추정하지 않는다.

[그림 8]은 모션 추정 단계의 결과 영상을 보여준다. [그림 8]의 (a)는 영상 시퀀스의 현재 프레임이고, (b)는 추정된 모션 필드를 나타낸다. [그림 8]의 (b)에서는 두 종류의 모션이 존재하는데 하나는 배경모션이고, 다른 하나는 굵은 선으로 표현된 내부의 모션들로 표현된 겹쳐진 영역의 모션들이다. 또한, 회색영역은 겹쳐진 영역을 표현하고, 이 영역들의 모션은 주변 블록들의 모션으로부터 추정된다. 예시된 영상에는 문자모션은 존재하지 않는다.



(a)



(b)

그림 8. 모션추정의 결과; (a) 현재 프레임, (b) 모션 추정의 결과
 Fig. 8. Example of motion estimation; (a) current frame, (b) result of motion estimation.

3. 시공간적 복원

인위적으로 영상에 삽입된 문자들이 여러 프레임동안 고정된 위치에 존재할 경우, 또는 인위적으로 영상에 삽입된 문자가 아닌 영상 자체에 포함된 문자들의 경우에는, 모션정보가 부재하므로 공간적 복원 알고리즘만을 사용하여 겹쳐진 영역을 복원해야 한다. 또한 문자 영역들의 생긴 모양이나 위치가 사전에 알기 힘들고, 영상 시퀀스 상에서의 글자 영역 주변의 텍스처가 손상되어 있는 경우가 대부분이기 때문에 공간적 복원만으로는 겹쳐진 영역을 복원하기가 어렵다. 따라서 본 논문에서는 시간적인 복원 방법과 공간적인 복원 방법을 결합하여 사용한다. 하지만, 앞서 언급한 두 경우가 아니라면 배경영역의 모션정보와 문자모션 정보를 활용하여 문자에 의해 겹쳐진 영역인 배경영역을 시간적 복원과 공간적 복원방법이 결합된 형태로 영역 복원을 수행한다.

입력된 연속 영상이 네 가지 Type 중에 하나로 결정되면, 각 시퀀스의 형태들에 적합한 시공간적 영상복원 방법이 적용된다. 제안된 방법은 연속 영상 시퀀스 상에서의 시간적 복원을 수행한 후, 복원이 되지 않은 나머지 픽셀에 대해 공간적 복원 방법을 적용한다. 제안된 시공간적 복원방법은 수식(9)과 같이 시간적 복원방법과 공간적 복원 방법의 가중치가 부가된 합의 형태로 구성된다.

$$\hat{I}(i, j, t) = \alpha \hat{I}_T(i, j, t) + (1 - \alpha) \hat{I}_S(i, j, t) \quad (9)$$

여기서 α 는 가중치 계수이고, $\hat{I}_T(i, j, t)$ 와 $\hat{I}_S(i, j, t)$ 는 각각 t 시간에 (i, j) 위치의 픽셀값에 대한 시간적 복원 결과와 공간적 복원 결과이다. 특히 시간적 정보를 이용한 복원에서는 입력 시퀀스를 분류하기 위하여 문자모션 정보와 배경모션 정보를 사용한다. 모션 추정결과를 이용한 시간 t 에서의 시간적 복원은 아래 수식(10)에 의해 수행된다. 시간적 복원을 수행할 때, 겹쳐진 영역들이 인접한 블록들과 같은 모션을 가진다고 가정하여 실제 겹쳐진 영역들의 모션을 인접한 8개의 배경영역 블록들의 모션정보에 대한 평균치로 추정하였다. 이때, 문자 픽셀을 포함한 블록들은 제외되었다. 시간적 복원과 공간적 복원의 비율을 계산하는 수식(9)에서 α 는 시공간적 복원에서 (i, j) 위치의 픽셀이 시간적 복원으로 복원되었는다면 1이고, 그렇지 않다면 0으로 설정한다.

$$\hat{I}_T(i, j, t) = I(i + d_1^i + d_1^b, j + d_2^i + d_2^b, t - 1). \quad (10)$$

수식(10)에서 픽셀 $(i + d_1^i, j + d_2^i) \notin TextArea(i, j, t - 1)$ 이다. $TextArea$ 는 각 프레임에서 겹쳐진 영역(문자영역)을 표현하고, $TextArea(i, j, t - 1) = TextArea(i + d_1^i, j + d_2^i, t)$. $I(i, j, t - 1)$ 는 $t - 1$ 시간에 (i, j) 위치의 픽셀값 이고, d^i 와 d^b 는 각각 (i, j) 위치의 픽셀을 포함하는 문자모션벡터와 배경모션 벡터이다.

겹쳐진 영역에서 시간적 복원의 수행으로 복원되지 못한 겹쳐진 픽셀들을 복원하기 위하여 공간적 복원 알고리즘으로 인페인팅 알고리즘을 수행한다. 인페인팅 알고리즘은 Bertalmio et al.에 의해 제안된 방법으로 영상의 손상된 부분을 복원하는 방법이고, 선택된 영역의 주변정보를 이용하여 복원할 영역을 자동으로 복원해주는 방법이다^[2]. 인페인팅 알고리즘의 기본적인 생각은 인페인팅되어야 할 영역 주변의 정보들이 그 영역의 경계에 도달하는 등조선(isophoto line)을 따라 그 영역의 안쪽으로 주변 정보를 전파해가면서 복원하자는 것이다. 인페인팅 알고리즘의 n 번째 반복에서 영상의 한 픽셀은 아래 수식(11)와 같이 갱신된다.

$$I^{n+1}(i, j) = I^n(i, j) + \Delta t I_n^*(i, j). \quad (11)$$

수식(11)에서 $I^n(i,j)$ 는 인페인팅될 영역의 안쪽에 있는 각각의 픽셀들을 의미하고, $\Delta t I^n(i,j)$ 는 t 시간의 픽셀 $I(i,j)$ 가 n 번째 반복 시에 Δt 의 비율로 갱신될 량을 표현한다. 이 갱신될 량을 표현하는 항은 전파해야 할 정보를 표현할 뿐 아니라, 2차원 라플라시안 연산자(2-D Laplacian operator)를 사용한 등조선의 방향까지 포함하고 있다. 수식(12) 처럼 영상의 에지정보를 유지하고, 노이즈에 대한 민감성을 제거하기 위하여 공간적 복원을 수행하는 동안 10번 반복마다 한번씩 anisotropic diffusion을 수행한다^[2,28].

$$\frac{\partial}{\partial t} I(x,y,t) = g(\nabla I) \kappa(x,y,t) |\nabla I(x,y,t)|, \mathbf{V}(x,y) \in \Omega. \quad (12)$$

수식(12)에서 $g(\nabla I) = e^{-(|\nabla I|/K)^2}$ 이고, $\kappa(x,y,t)$ 는 등조선의 유클리디언 곡률(Euclidean curvature)을 나타낸다. 그리고 ∇I , Ω , K 는 각각 영상의 기울기, 인페인팅을 수행할 영역 그리고, 표준화 상수(normalizing constant)를 나타낸다. 공간적 복원 알고리즘은 겹쳐진 영역 내의 픽셀값들이 더 이상 변하지 않을 때까지 수행된다. 자세한 사항은 참고논문[2]를 참고하고, 다음은 인페인팅 알고리즘의 pseudo code이다.

Pseudo code of Inpainting Algorithm

1. 전처리 (anisotropic diffusion)
2. 해당영역의 각 픽셀에 대하여,
 - 2.1. 2D Laplacian 연산자를 이용한 수직 수평방향의 smoothness를 계산
 - 2.2. 인페인팅 방향 결정
 - 2.3. 등조선의 방향을 따라 계산된 2D smoothness를 전파
 - 2.4. 각 픽셀위치의 magnitude 계산
 - 2.5. 2.1~2.4에서 계산된 값을 이용한 픽셀값 갱신
3. 영역의 smoothness가 평형상태가 될 때까지 2.1에서 2.5를 반복 수행

IV. 실험 결과

제안된 방법은 여러 개의 영화와 애니메이션으로부터 수집된 비디오 클립을 사용하여 실험하였다. 실험 동안 사용된 영상은 352×240 픽셀부터 720×400 픽셀 크기이며, 비디오 클립으로부터 초당 15장의 프레임을 캡처하여 사용하였다. 각 프레임에서 문자의 크기는 6×6픽셀부터 35×36픽셀이며, 각 글자의 폰트 넓이는 2픽셀부터 5픽셀까지이다.

제안된 방법은 Visual C++를 사용하여, P-4 1.7 GHz 사양의 시스템에서 실험되었다. SVM의 훈련을 위하여 영화나, TV 프로그램, 애니메이션과 같은 다양한 영상들로부터 수동으로 캡처된 다양한 크기들의 150개의 글자가 포함되어 있는 영상들을 사용하였다. 훈련영상에서 문자 부분의 $M \times M$ 윈도우가 양성패턴으로써 사용되었고, 문자가 아닌 영상의 나머지 부분들이 음성패턴으로써 사용되었다. 시간적 정보를 활용할 수 없는 경우에는 공간적 정보만을 사용하여 겹쳐진 영역들을 복원해야 한다. 이때 일반적으로 압축된 영상들은 압축률이 높을수록 영상에 포함된 객체들에 관련된 많은 정보를 잃어버리기 때문에 겹쳐진 영역을 둘러싸고 있는 주변 영역의 공간적 정보는 손상되고, 공간적 복원에서 신뢰할 만한 결과를 얻기가 힘들다. 동적배경 상의 고정문자 시퀀스에 대한 실험에서, 배경이 동적으로 움직이기 때문에 시간적 정보는 겹쳐진 영역의 복원에 유용하다. 시공간적 복원은 겹쳐진 영역들이 시간적 복원에서 부분적으로 복원될 수 있기 때문에 대상 영역들의 크기가 좀더 작아지며, 공간적으로 복원된 픽셀들 주위의 공간적 정보가 보다 정확하기 때문에 공간적 복원보다 더 좋은 결과를 보여준다. 현재의 프레임에서 카메라나 물체의 움직임에 의해 발생된 새로이 드러나는 배경영역은 시간적 복원 시에 대상영역이 된다.

[그림 9]은 동적배경상의 정지문자 영상에 대한 실험 결과를 보여준다. [그림 9]의 (a)와 (b)는 640×272 크기의 현재와 이전 프레임이다. [그림 9]의 (c)는 (a)와 (b)사이의 차이 영상이고, 이 영상에서 흰 픽셀들은 두 영상의 같은 위치의 픽셀값이 0이 아닌 부분을 나타낸다. [그림 9]의 (d)는 적응적 임계치 알고리즘에 의해 생성된 임계값 21이 적용되었을 때의 차이 영상이다. 즉, 흰색 픽셀은 21보다 크거나, -21보다 작은 차이를 보인 픽셀들을 나타낸다. [그림 9]의 (e)는 확장된 글자영역 나타내고, 작게 분리된 영역들은 SVM기반의 문자 검출기에 의한 오류들이다. [그림 9]의 (f)는 시간적 복원에 대한 대상 영역들을 보여준다.

[그림 9]의 (f)영상에서 진회색(dark gray)으로 표시된 영역은 인접한 주변 블록들의 모션이 존재하지 않거나, 해당 블록 자체의 모션이 존재하지 않는 영역을 의미한다. 그리고, 연회색(light gray)으로 표시된 영역은 시간적 복원의 대상 영역들으로써 논리적으로 복원 가능한 영역이면서 추정된 모션이 존재하는 영역을 나타낸다. [그림 9]의 (g)와 (h)는 시간적 복원 결과와 시공간적 복원



그림 9. 시공간적 복원의 결과: (a) 현재 프레임, (b) 이전 프레임, (c) 프레임 차이 영상, (d) 적응적 임계값을 적용한 프레임 차이 영상, (e) 겹쳐진 영역들, (f) 시간적으로 복원가능한 영역들, (g) 시간적 복원의 결과, (h) 시공간적 복원의 결과.
 Fig. 9. Spatiotemporal restoration results: (a) current frame, (b) previous frame, (c) frame difference image, (d) frame difference image after applying adaptive threshold technique, (e) occluded regions, (f) the temporally restorable regions, (g) result of temporal restoration, and (h) final result of spatiotemporal restoration.

의 최종 결과를 보여준다. [그림 9]의 (g)나 (h)와 같이 시간적 복원에 의해 복원될 픽셀들의 위치가 이전의 프레임에서 여전히 글자 영역에 포함되어 있기 때문에, 논리적으로 복원 가능한 영역들이 시간적 복원 알고리즘을 수행한 후에도 완전히 복원 되지 않았다. 따라서 논리적으로 복원 가능한 영역이라 할지라도 실제로 모두 복원 되는 것은 아니다.

[그림 10]과 같이 동적배경상의 동적문자 영상에 대한

실험에서 겹쳐진 영역은 두 모션정보(문자모션과 배경모션)를 이용하여 대상영역을 복원하다. 이 경우에는 문자의 움직임으로 인해 새로이 겹쳐진 영역과 배경모션에 의해 새로이 드러난 영역 모두가 대상영역이 되고, 두 모션정보를 이용하여 복원된다. 즉, 다른 모든 경우보다 동적배경상의 동적문자 시퀀스의 경우가 시간적 복원으로 복원 가능한 영역의 범위가 제일 넓다. [그림 10]은 시공간적 복원의 결과를 보여준다. [그림 10]의 (a)와 (b)는



그림 10. 시공간적 복원의 결과: (a) 현재 프레임, (b) 이전 프레임, (c) 확장된 문자 영역, (d) 시간적 복원 알고리즘으로 복원 가능한 영역, (e) 시간적 복원의 결과, (f) 최종 결과.

Fig. 10. Results of the spatiotemporal restoration: (a) current frame, (b) previous frame, (c) result of text expanding, (d) temporally restorable areas, (e) result of temporal restoration, and (f) final result of spatiotemporal restoration.

384×288크기의 현재 프레임과 이전 프레임 영상이고, (c)는 확장된 문자영역의 결과이다. [그림 10]의 (c)에서 좌측상단부분의 글자들은 휴리스틱 규칙들을 만족하지 못하기 때문에 제거되었다. [그림 10]의 (d)는 흰색으로 표현된 복원 가능한 영역들이 배경모션과 문자모션정보 모두를 사용하여 복원될 영역을 표현한다. 연회색 영역들은 문자모션만을 사용하여 복원될 영역을 표현한다. 시공간적 복원방법을 이용한 결과영상 [그림 10]의 (e)에서 보이는 것처럼 시간적인 복원없이 공간적 복원만으로 겹쳐진 영역을 복원했다면 현재 프레임의 입술부분의 복원은 불가능하다.

V. 제안된 방법의 평가 및 고찰

본 장에서는 제안된 방법의 성능 평가를 위해 사용된 인접한 두 프레임 영상을 소개하고, 이 영상을 이용하여

표 2. 시퀀스 분류의 정확도
Table 2. Accuracy of sequence classification.

입력 영상 시퀀스 Type	프레임 쌍의 수 (비디오 클립의 수)	오분류의 수	정확도 (%)
Type 1	514 (5)	18	96.5
Type 2	1150 (12)	83	92.78
Type 3	1828 (8)	102	95.3
Type 4	1382 (6)	0	100
전체	4874 (31)	203	95.84

4가지 형태의 문자가 포함된 영상 시퀀스를 구성한 후, 시공간적 복원 방법과 공간적 복원방법에 대한 결과를 비교 설명한다.

제안된 방법의 평가는 크게 4가지 측면에서 이루어진다. 첫째, 4가지 Type의 입력 시퀀스 분류의 정확도(Sequence classification accuracy), 둘째, 시공간적 복원 방법과 공간적 복원 방법의 처리시간, 셋째, 복원된 영상과 원 영상을 비교하기 위한 수단으로 ISNR (Improvement of Signal-to-Noise Ratio)을 척도하여 설명한다 [29-30]. 마지막으로 주어진 입력 시퀀스에서 제안된 방법과 기존의 방법간 적용가능성과 그 장단을 비교 설명한다.

1. 시퀀스 분류의 정확도 평가

시퀀스 분류의 정확도를 측정하기 위해 전체적으로 31개의 비디오 클립으로부터 4874 개의 프레임 쌍을 실험하였다. 그 결과 95.84%의 시퀀스 분류 정확도를 보였다.

오분류(classification error)를 분석한 결과, 대부분의 오류는 배경의 움직임 판단하는 과정과 글자 픽셀 분류단계에서의 오류에 기인한다. 전체적인 오류의 종류와 원인은 아래와 같다.

- Type 1: 부적절한 임계값 (T_2)의 설정으로 인해 배경에 움직임이 없음에도 불구하고 배경이 움직인 것으로 판단되어 Type 2 이미지 시퀀스로 분류되었고, 또한영상의 압축과정에서 텍스트의 위치가 왜곡됨으로 인해 글자가 움직인 경우인 Type 3 이미지 시퀀스로 분류된 경우가 발행하였다.

- Type 2: Type 2 이미지 시퀀스는 배경영역에만 움직임이 있지만, 문자영역의 검출단계에서의 오류로 인하여 문자영역으로 잘못 분류된 배경영역의 움직임에 의해, 문자영역에 움직임이 있는 것으로 잘못 분류되었다. 그 결과 Type 4 이미지 시퀀스로 잘못 분류된 경우가 발행하였다.

- Type 3: Type 3 이미지 시퀀스는 문자영역에만 움직임이 있는 시퀀스이지만, 문자영역의 검출단계에서의 오류로 인하여 문자영역을 검출하지 못한 경우, 문자영역의 움직임으로 인해 배경영역이 움직인 것으로 잘못 분류되어 결과 Type 4 이미지 시퀀스로 잘못 분류된 경우가 발행하였다.

- Type 4: Type 4 이미지 시퀀스의 경우는 오분류가 발행하지 않았다.

[그림 11]은 제안된 방법의 성능을 평가하기 위해 구성된 4가지 형태의 영상 시퀀스에 사용된 현재와 이전 프레임 영상과 시간적 복원의 중간결과 영상들이다. 640×352 해상도의 현재와 이전 영상으로부터, 수작업으로 문자열을 삽입하여 4가지 형태의 입력 영상 시퀀스를 구성하였다. 평가를 위한 실험으로 각 Type의 입력 영상 시퀀스를 제안된 방법(시공간적 문자영역 제거)으로 문자 영역을 제거하였다. 문자의 모션을 위해 합성 이미지 시퀀스에서 현재 프레임에서 위쪽 방향으로 5픽셀을 옮겨 삽입하여 실험하였다.

[그림 11]의 (c), (d), (e)와 (f)는 각각 고정배경 상의 고정문자 시퀀스(Type 1 시퀀스), 동적배경 상의 고정문자 시퀀스(Type 2 시퀀스), 고정배경 상의 동적문자 시퀀스(Type 3 시퀀스), 동적배경 상의 동적문자 시퀀스(Type 4 시퀀스)에 대하여 시간적 복원의 중간 결과를 보여준다.

[그림 12]는 시간적 복원 이후에 공간적 복원 방법으로 제거할 문자영역의 픽셀 수를 계산하여 도시한 그래프이다. [그림 12]에서 보듯이 시간적 복원 이후의 공간적 복원으로 배경영역을 복원해야하는 픽셀의 수가 Type 2, 3, 4 이미지 시퀀스의 경우, 42495 픽셀에서 각각 21123, 17825, 22178로 줄었다. 따라서 공간적 복원에 소요되는 시간상의 감소효과를 볼 수 있다. 또한 시간적 복원을 수행함으로써 공간적 복원 시 주변의 텍스처적인 정보를 보강할 수 있기 때문에 좀더 정확한 공간적 복원

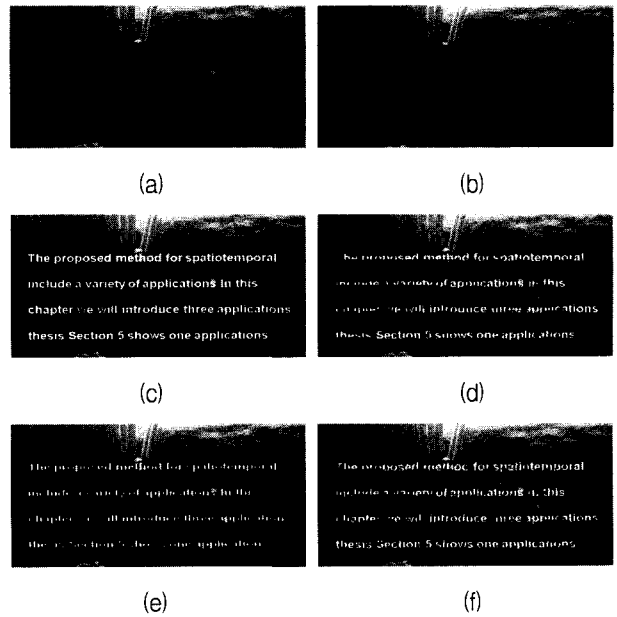


그림 11. 시간적 복원의 중간결과 영상; (a) 현재 영상, (b) 이전 영상, (c) Type 1, (d) Type 2, (e) Type 3, (f) Type 4.

Fig. 11. Intermediate results of temporal removal of text (a) current frame, (b) previous frame, (c) Type 1, (d) Type 2, (e) Type 3, (f) Type 4.

결과를 획득할 수 있다는 장점이 있다.

2. 처리시간에 관한 고찰,

본 절에서는 제안된 방법을 [그림 11]의 (a)와 (b)영상 (640×352)에 적용한 실험결과와 공간적 복원 방법(Bertalmio *et al.*^[2])으로 실험한 결과를 비교 설명한다.

[표3]에서 픽셀분류는 SVM을 이용한 문자 픽셀 분류에 소요된 전체 시간을 의미하고, 시퀀스 분류는 입력 이미지 시퀀스가 어느 Type에 속하는 지를 결정하는 단계에서 소요된 시간이다. 또한 시간적 복원은 이전 프레임으로부터 시간적 복원을 수행하기 위해 소요된 시간이고, 공간적 복원은 참고문헌 [2]에서 제안된 방법으로 공간적 복원을 수행하기 위해 소요된 시간이다. [표3]에서 보듯이, 전체적인 시공간적 복원의 수행시간이 각 Type 1, 2, 3, 4 이미지 시퀀스에 따라 0초, 134.56초, 최대 156.12초, 127.19초의 감소효과를 보였다. Type 1 이미지 시퀀스의 경우, 고정배경 상의 고정문자 시퀀스이기 때문에 시간적인 정보의 부재로 시간적 복원 단계에서 전혀 복원이 발생하지 않았다. 공간적 복원에서 소요되는 시간이 시공간적 복원에 소요되는 전체시간의 대부분을 차지하기 때문에, 제안된 방법을 실 응용분야에 적용하

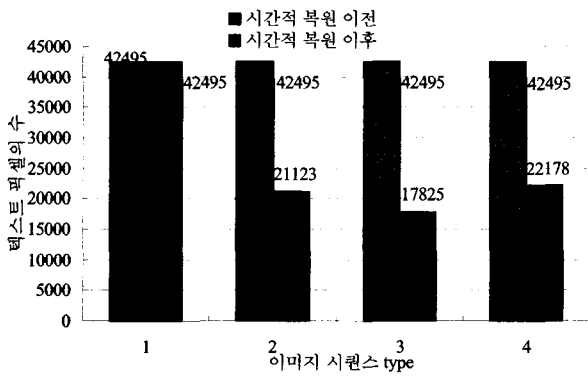


그림 12. 시간적 복원 후의 텍스트 픽셀의 수.
Fig. 12. The number of text pixels before and after the temporal removal.

기 위해서는 에 시간상의 오버헤드(Overhead)를 해결해야만 한다. 따라서 계산상의 효율을 보기위해 좀더 빠른 공간적 복원방법이 필요하고, 최근 Oliveira *et al.*^[31]에 의해 제안된 “Fast Digital Inpainting Algorithm”이 그 대안들 가운데 한가지라 할 수 있을 것이다.

3. 복원의 정확도 평가

복원의 정확도 평가를 위하여 공간적인 복원 방법들 중에 하나인 Bertalmio *et al.*의 방법^[2]과 식(13)을 이용하여 평가하였다.

$$ISNR=10 \cdot \log_{10} \left(\frac{\sum_{i,j} (f(i,j) - g(i,j))^2}{\sum_{i,j} (f(i,j) - \hat{f}(i,j))^2} \right) \quad (13)$$

여기서, $f(i, j)$ 는 원 영상, $\hat{f}(i, j)$ 는 복원된 영상, 그리고 $g(i, j)$ 는 관측된 영상을 의미한다^[30]. 비록 ISNR을 이용한 복원의 정확도 척도가 복원의 전체적인 성질을 반영하지 않고, 이미지 질의 향상을 전적으로 반영하지는 않지만, 복원결과의 질의 향상을 척도하기위한 목적으로는 사용할 만한 척도이다.

제안된 방법의 척도를 위해 각 RGB 칼라 공간의 각 평면(plane)에서 그 질의 향상을 척도 하였다. [표4]에서 보듯이, 복원결과의 질의 향상을 척도한 결과, Bertalmio *et al.*의 방법^[2]보다 세 칼라 평면에서 각 Type 2, 3, 4 이미지 시퀀스에 대하여 0.2dB, 4.9dB, 2.0dB 의 향상을 보였다.

4. 적용 가능성(Availability)에 대한 고찰

본 절에서는 [표5]에서 보이는 바와 같이 제안된 방법

표 3. 처리시간의 비교.

Table 3. Comparison result of processing time(sec).

시퀀스 Type	픽셀 분류	시퀀스 분류	시간적 복원	공간적 복원	총소요시간	
					시공간적 복원	공간적 복원
1	3.24	0.30	N/A	269.62	273.16	273.16
2	3.24	0.30	1.04	134.02	138.60	273.16
3	3.24	0.30	0.40	113.10	117.04	273.16
4	3.24	0.30	1.72	140.71	145.97	273.16

표 4. 복원의 정확도 비교

Table 4. Comparison result of ISNR(dB).

	Plane	Type 1	Type 2	Type 3	Type 4
Bertalmio <i>et al.</i> 의 방법 ^[2]	Red	24.9621	24.9621	24.9621	24.9621
	Green	25.0312	25.0312	25.0312	25.0312
	Blue	21.771	21.771	21.771	21.771
	평균	23.9214	23.9214	23.9214	23.9214
제안된 방법	Red	24.9621	25.6694	29.6346	27.6363
	Green	25.0312	25.2988	30.227	26.8436
	Blue	21.771	21.5915	26.4682	23.3809
	평균	23.9214	24.1866	28.7766	25.9536

표 5. 적용 가능성 비교

Table 5. Availabilities of removal method.

방법	Type 1	Type 2	Type 3	Type 4
Bertalmio <i>et al.</i> 의 방법 ^[2]	0	0	0	0
Wei와 Levoy의 방법 ^[4]	0	0	0	0
Irani와 Peleg의 방법 ^[5]	×	×	0	0
Chun과 Bae의 방법 ^[6]	×	0	×	0
제안된 방법	0	0	0	0

과 기존의 방법들이 이미지 시퀀스 상에서 문자영역을 제거하고, 그 배경영역을 복원하기 위한 적용 가능성과 각 방법의 장단을 설명한다.

먼저 Bertalmio *et al.*의 방법^[2]은 모든 Type의 입력 이미지 시퀀스에 적용이 가능하지만, 공간적 정보를 사용할 수 없기 때문에, 계산상의 오버헤드와 복원결과의 정확도면에서 신뢰할 만하지 않다는 단점이 있다. 또한, 이 방법은 복원해야할 영역의 내부에 존재하는 세부사항

들은 복원 불가능하다. Wei와 Levoy의 방법^[4]은 텍스처 합성으로 손실된 영역을 복원하려는 시도이기 때문에, 사용자가 복원하고자 하는 영역에 맞는 텍스처를 지닌 영역을 탐색해야하고, 해당부분을 분할한 다음, 복원할 영역에 맞게 합성하여 복원하는 작업이 필요하다. 따라서 상당히 많은 오버헤드가 존재하고, 특히, 복원하고자 하는 영역과 유사한 텍스처를 지닌 영역이 항상 존재한다는 보장을 받기가 어렵다. Irani와 Peleg의 방법^[5]는 겹침 제거(occlusion removal)에 초점을 맞추었기 때문에, 문자영역의 모션이 그 문자가 가리고 있던 배경영역이 들어날 때 까지 충분한 모션이 존재해야하고, 또한, 고정된 문자 시퀀스의 경우 전혀 배경영역을 복원할 수 없다는 단점이 존재한다. 따라서 Irani와 Peleg의 방법은 이미지 시퀀스 상에서 문자영역의 제거에 적합하지 않은 방법이다. Chun과 Bae의 방법^[6]은 배경영역의 모션에 초점을 맞추어 제안한 방법이므로, 문자가 가리고 있던 배경영역을 복원하기 위해 충분한 배경모션이 존재해야하고, 배경영역이 움직이지 않을 경우에는 문자영역이 가리고 있던 배경영역을 복원할 수 없다. 앞서 언급한 기존의 방법들에 비해 제안된 방법은 4가지 Type의 이미지 시퀀스에 모두 적용이 가능한 방법이고, 기존방법에 비해 많은 장점이 있다. 제안된 방법은 입력 이미지 시퀀스 상에서 배경영역의 객체와 문자의 특징을 파악하여 입력 영상을 4가지 형태로 분류하여 각 Type에 맞는 복원 알고리즘을 수행함으로써, 보다 정확한 복원의 결과를 획득할 수 있었다.

VI. 결론

본 논문에서는 비디오나 영상 자료의 원 영상(문자 정보가 없는 영상)을 복원하기 위해 시공간 정보를 이용한 문자영역 제거 방법을 제안하였다. 제안된 방법은 전 과정을 자동화하기 위하여 SVM을 이용한 문자 검출 기술과 검출된 문자 영역에 의해 겹쳐진 영역의 제거와 원 영상의 복원을 위하여 시공간적 복원 방법을 사용하였다. 시공간적 복원은 문자에 의해 가려진 영역을 모션 정보를 이용한 시간적 복원과 가려진 영역의 공간적으로 이웃한 주변 정보를 이용한 공간적 복원이 결합된 형태의 복원 방법이다. 제안된 방법의 실험결과에서 시공간적 복원 방법이 공간적 복원 방법보다 계산상의 효율이나, 결과영상이 원 영상과 보다 유사하게 복원됨을 입증하였다.

제안된 방법은 다음과 같은 몇 가지 장점이 있다. 1) 시간적인 복원을 먼저 수행함으로써, 복원 대상 영역의 일부분이 복원 가능하였으며, 그 결과로 인하여 공간적인 복원에 소요되는 계산 시간을 줄일 수 있다. 2) 시간적인 복원의 결과를 공간적인 복원에 사용함으로써, 공간적인 정보의 증대와 복원의 정확도를 높일 수 있다. 3) 입력 시퀀스의 특징과 관심의 대상이 되는 영역의 특징들을 조사하여, 입력 시퀀스를 4종류로 분류하였고, 그 특징에 맞는 복원 알고리즘의 수행으로 인한 시간의 효율성과 복원결과의 정확도를 높일 수 있다. 4) 시간적 복원이나, 공간적, 혹은 시공간적 복원이 모두 가능한 방법이다. 즉, 시공간적 정보 중 어느 하나의 부재 시에도 관심의 대상이 되는 영역을 복원할 수 있다. 5) MPEG과 같은 압축된 영상에서도 시간적인 복원으로 인하여 문자영역 주변의 텍스처 정보를 이전 영상에서 보충할 수 있기 때문에 시공간적인 복원을 사용하여 신뢰할 만한 복원결과를 구할 수 있다.

제안된 방법의 향후 과제는 다음과 같은 고찰에 기반을 두어 전개될 것이다. 첫째, 제안된 방법에서 수행시간의 대부분이 공간적 복원의 수행에서 소모된다. 일반적으로 복원될 영역의 텍스처와 유사한 텍스처가 동일한 영상 내에 많이 존재하기 때문에, 제안된 방법의 수행 속도 향상을 위해 텍스처 합성과 같은 방법이 고려될 것이다. 둘째, 제안된 방법에서는 글자의 색깔에 대한 가정을 포함하고 있다. 하지만, SVM의 문자 검출 능력에 비해, 글자의 색깔에 대한 가정이 엄격하기 때문에 다양한 색깔의 문자를 추출하기 위한 방법이 고려될 것이다. 셋째, 연속 입력 시퀀스에 대한 고려로써, 이전 시퀀스에서 검출된 문자 영역에 대한 추적을 고려한다면, 문자 검출에 소요되는 시간을 줄일 수 있을 것이다. 따라서 문자 영역의 추적에 관한 연구를 고려 중이다. 넷째, 모션 보상에 관한 연구결과를 토대로 시간적인 복원의 정확도 향상을 위한 방안이 고려될 것이다. 마지막으로 제안된 방법의 응용분야 중, 변형 가능 템플릿을 이용한 ROI의 자동 검출 및 제거 방법에 관한 연구를 진행할 것이다.

참고 문헌

- [1] A. C. Kokaram, R. D. Morris, W. J. Fitzgerald, P. J. W. Rayner, "Interpolation of Missing Data in Image Sequences," *IEEE Transaction on Image Processing*, vol. 4, no. 11, pp. 1509~1519, 1995.
- [2] M. Bertalmio, G. Sapiro, V. Caselles, C. Ballester, "Image Inpainting," *Siggraph 2000 Conference Proceedings*, pp. 417~424, 2000.
- [3] T. Chan, J. Shen, "Inpainting, zooming, and edge coding," *Special Session on Inverse Problems and Image Analysis at the AMS Annual Conference, January 2001*.
- [4] L. Y. Wei, M. Levoy, "Fast Texture Synthesis using Tree-structured Vector Quantization," *Siggraph 2000 Conference Proceedings*, pp. 479-488, 2000.
- [5] M. Irani, S. Peleg, "Motion Analysis for Image Enhancement: Resolution, Occlusion, and Transparency," *Journal on Visual Communications and Image Representation*, vol. 4, no. 4, pp. 324~335, 1993.
- [6] B. T. Chun, Y. Bae, "A Method for Recovering Original Image for Video Caption Area and Replacing Caption Text," in *Proceeding on International Workshop of Content-based Multimedia Indexing'2001 (CBMI'01)*, Brescia, Italy, September, 2001.
- [7] R. Lienhart, F. Stuber, "Automatic Text Recognition in Digital Videos," *SPIE-The International Society for Optical Engineering*, pp. 180-188, 1996.
- [8] A. K. Jain, B. Yu, "Automatic Text Location in Images and Video Frames," *Pattern Recognition*, vol. 31, no. 12, pp. 2055~2076, 1998.
- [9] E. Y. Kim, K. Jung, K. Y. Jeong, H. J. Kim, "Automatic Text Region Extraction Using Cluster-based Templates," *International Conference on Application and Pattern Recognition and Digital Techniques*, pp. 418~421, 2000.
- [10] M. D. Richard, R. P. Lippmann, "Neural Network Classifiers Estimates Bayesian a Posteriori Probabilities," *Neural Computation*, vol. 3, pp. 461~483, 1991.
- [11] Y. Zhong, K. Karu, A. K. Jain, "Locating Text in Complex Color Images," *Pattern Recognition*, vol. 28, no. 10, pp. 1523~1535, 1995.
- [12] A. K. Jain, K. Karu, "Learning Texture Discrimination Masks," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 18, no. 2, pp. 195~205, 1996.
- [13] K. Y. Jeong, K. Jung, E. Y. Kim, H. J. Kim, "Neural Network-based Text Location for News Video Indexing," *Proc. of International Conference of Image Processing*, vol. 3, pp. 319~323, 1999.
- [14] H. Li, D. Doerman, O. Kia, "Automatic Text Detection and Tracking in Digital Video," *IEEE Transactions on Image Processing*, vol. 9, no. 1, pp. 147~156, 2000.
- [15] Y. Zhong, H. Zhang, A. K. Jain, "Automatic Caption Localization in Compressed Video" *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 4, pp. 385-392, 2000.
- [16] Y. M. Y. Hasan, L. J. Karam, "Morphological Text Extraction from Images," *IEEE Transactions on Image Processing*, vol. 9, no. 11, pp. 1978~1983, 2000.
- [17] K. Jung, "Neural network-based text location in color images," *Pattern Recognition Letters*, vol. 22, pp. 1503~1515, 2001.
- [18] T. Hastie, R. Tibshirani, J. Friedman, *Elements of Statistical Learning: Data Mining, Inference and Prediction*, Springer-Verlag, New York, 2001.
- [19] K. I. Kim, K. Jung, S. H. Park, H. J. Kim, "Support Vector Machines for Texture Classification," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24,

- no. 11, pp. 1542~1550, 2002.
- [20] B. Schölkopf, A. J. Smola, Learning with Kernels, The MIT Press, 2002.
- [21] T. M. Cover, "Geometrical and Statistical Properties of Systems of Linear Inequalities with Applications in Pattern Recognition," IEEE Transactions on Electronic Computers, vol. 14, pp. 326~334, 1965.
- [22] V. Vapnik, Statistical Learning Theory, John Wiley & Sons, New York, 1998.
- [23] Schölkopf B., Sung K. K., Burges C. J. C., Girosi F., Niyogi P., Poggio T., and Vapnik V., "Comparing Support Vector Machines with Gaussian Kernels to Radial Basis Function Classifiers," IEEE Transactions on Signal Processing 45, pp. 2758~2765, 1997.
- [24] Haykin S. Neural Network-A Comprehensive Foundation, 2nd edition. Prentice Hall, NJ.1999.
- [25] S. Schölkopf, C. J. C. Burges, V. Vapnik, "Extracting Support Data for a Given Task," Proc. Int. Conf. on Knowledge Discovery & Data Mining, pp. 252~257, 1995.
- [26] A. Bovik, Hand Book of Image and Video Processing, Academic Press, 2000.
- [27] N. Habili, A. Moini, N. Burgess, "Automatic Thresholding for Change Detection in Digital Video," Proc. Visual Com. And Image Processing, 4067, pp. 133~142, 2000.
- [28] P. Perona, J. Malik, "Scale-space And Edge Detection using Anisotropic Diffusion," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 12, no. 7, pp. 629~639, 1990.
- [29] C. W. Lee, Spatiotemporal Approach to Removal of Superimposed Text in Image Sequences, PhD thesis, Kyungpook National University, 2003. 12.
- [30] R. Nakagaki, and A. K. Katsaggelos, "A VQ-Based Blind Image Restoration Algorithm," IEEE Transactions on Image Processing, vol. 12, no. 9, pp. 1044~1053, 2003.
- [31] M. M. Oliveira, B. Bowen, R. McKenna, and Y. S. Chang, "Fast Digital Image Inpainting," in Proc. of the International Conference on Visualization, Imaging and Image Processing, pp. 261~266, 200.

— 저 자 소 개 —



이 창 우(정회원)

1998년 경북대학교
컴퓨터공학과 공학석사
1999년~2001년 포항1대학 교수,
2004년 경북대학교 컴퓨터공학과
공학박사

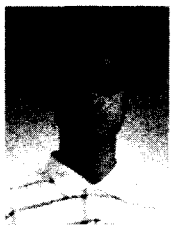
<주관심분야 : 비전인식, 패턴 인식, 비디오 인
텍싱, Video Retrieval, 문자 검출>



강 현(정회원)

1999년 경북대학교
컴퓨터공학과 공학석사
2000년~2001년 (주)네비웍스
책임연구원
2004년 경북대학교 컴퓨터공학과
공학박사

<주관심분야 : 게임, HCI, Augmented Reality, 이
미지처리, 제스처인식, 문자인식>



정 기 철(정회원)

1996년 경북대학교
컴퓨터공학과 공학석사
2000년 경북대학교 컴퓨터공학과
공학박사
2001년~2002년 미국 미시간대
Anil K. Jain 교수 PRIP
연구실 박사 후 연구원,

2003년~현재 숭실대학교 미디어학부 교수
<주관심분야 : 문자 인식, 이미지 처리, 패턴 인
식, 비디오 인텍싱, Augmented Reality, Mobile
Vision System>

김 항 준(정회원)

제40권 CI편 제6호 참조