

구조적 특성을 갖는 Texture 영상의 해석

(The Analysis of Texture Images with Structural Characteristics)

申 在燮*, 朴來弘**

(Jae Seop Shin and Rae-Hong Park)

要 約

일반적으로 반복적인 형태를 가지는 텍스처(texture) 영상은 그들의 모양(shape) 해석을 통하여 가장 표준이 되는 텍스처 모델과 regularity vector에 의해 묘사될 수 있다. 기존의 해석 방법들은 regularity vector를 구하기 위해서 너무 많은 시간을 소비할 뿐만 아니라 계산도 복잡하다. 본 논문에서는 계산 과정이 매우 간단하고 계산 시간도 상당히 줄 수 있는 방법을 제안하였다. 또, 전처리과정에서 텍스처 영상에 적용할 수 있는 몇몇의 향상된 알고리즘의 적용을 시도하였다. 마지막으로 텍스처 해석의 결과를 이용하여 텍스처 영상을 합성함으로써 제안한 알고리즘의 유용성을 보였다.

Abstract

In general, texture images with regular patterns can be described by using the standard texture model and regularity vectors for their shape analysis. Early methods not only take much time but also have computational complexity in obtaining regularity vectors. The proposed method here simplifies computation process and saves computation time. And we proposed some improved preprocessing algorithms for texture analysis. Finally, we showed the utility of the proposed method through texture synthesis by making use of the results of texture analysis.

I. 序 論

텍스처(texture)란 물체의 구조 혹은 그들의 결합관계 등을 잘 나타내 주는 물체의 표면에 대한 조직 구조를 말하는데 여기에는 물체들 사이의 반복성이 없는 random 텍스처와 반복적인 형태를 가지는 regular 텍스처가 있다. Random 텍스처인 경우 통계적인 해석 방법이 많이 쓰이며 regular 텍스처인 경우 구조적인

해석 방법이 많이 사용된다.^{[1]~[5]}

구조적인 해석에서 가장 중요한 요소는 영상내에서 해석의 단위라고 할 수 있는 primitive의 추출과 primitive 상호간의 placement rule을 찾아내는 것이다. 입력 영상이 구조적으로 반복되는 텍스처 영상인 경우 이들의 텍스처 모델과 두 regularity vector에 의해 전체 영상을 묘사할 수 있으므로 영상을 전송할 경우 정보 압축의 효과가 크며 이런 영상이 한 영상내에 서로 다른 형태로 존재할 때 이들의 분할도 가능하며 송, 수신 측에서 미리 정해 놓은 정보 table에 의해 소량의 정보만을 이용하여 텍스처 영상을 합성할 수도 있는 장점을 가진다.

본 논문에서는 텍스처 영상이 갖고 있는 배열구조가 두 vector의 합과 차에 의해서 표현될 수 있는 구조여

*準會員, 三星綜合技術院 電子機器研究所
(Electronics Research Center, Samsung Advanced Institute of Tech.)

**正會員, 西江大學校 電子工學科
(Dept. of Elec. Eng., Sogang Univ.)
接受日字: 1987年 2月 18日

야 한다는 제한 조건을 전제로 하는 규칙적인 배열구조를 갖는 텍스춰 영상 해석에 관심의 초점을 두어 텍스춰 영상을 구성하고 있는 요소들의 특성 및 그들 사이의 공간적인 상호 관계를 잘 묘사해 줄 수 있는 구조적인 해석 방법에 대해 연구하였다며 더 나아가 random texture의 경우도 고려하였다. 또한 얇은 해석 결과를 가지고 텍스춰 합성을 통하여 텍스춰 해석 방법의 유용성을 보였다. II장에서는 본 연구를 위한 전처리 과정을 행하며 III장에서는 이진 영상을 사용하여 *textural feature* 및 *dominant primitive*를 찾아내는 과정에 대해서 다루고 IV장에서는 *dominant primitive*와 각 *primitive*들의 중심좌표를 이용하여 텍스춰 모델 및 공간 구조를 해석하며 V장에서는 VI장에서 얻은 텍스춰 모델과 공간 구조를 사용하여 텍스춰 영상을 합성하여 원래 영상과 비교를 함으로써 제안한 방법의 성능(*performance*)을 측정하며 VII장에서는 실험결과와 그에 따른 문제점을 살펴보고 VIII장에서는 실험을 토대로 한 결론을 내린다.

II. 전 처리

이 장에서는 텍스춰 해석을 위한 3 단계의 전처리 과정을 다루게 되는데, 첫째는 입력영상의 잡음 제거 및 국부적인 동일성을 높여주기 위한 평활화를 행하며 둘째는 이진 영상을 얻기 위한 임계 분할(*thresholding*) 과정을 거치게 된다. 마지막으로 임계분할의 오류를 보완해 주고 좀 더 깨끗한 이진 영상을 얻기 위한 연결된 *primitive*의 분리과정을 수행한다.

1. 가변 평활화(variable smoothing)

대부분의 입력 신호들은 잡음 성분을 포함하고 있으며 같은 물체임에도 불구하고 빛의 밝기가 균일하지 않기 때문에 잡음을 제거해야 하며 국부적인 동일성도 높여줄 필요가 있다. 일반적으로 각 weight가 1인 3×3 창(window)을 영상과 convolution 시킨 후 9로 나누어 주는 고정 평활화 기법을 사용하는데 인접한 화소들 사이에도 변화가 매우 심하거나 매우 섬세한 구조를 가지는 텍스춰 영상에 그대로 적용하면 서로 다른 영역에 영향을 주기 때문에 우리가 얻고자 하는 잡음제거 효과를 기대하기 어렵다. 그러므로 창내에 들어오는 영상의 조건에 따라 서로 다른 operation을 취해 주는 가변 평활화 기법을 사용해야 한다! [6]에서는 $2n \times 2n$ 창만을 고려했는데 그렇게 하면 test 화소의 값을 두는 위치에 따라 서로 다른 결과를 얻을 수 있을 뿐만 아니라 평활화된 값이 정확하지 않을 수도 있다. 그래서 본 논문에서는 test 화소의 값을 중심에 오도록 3×3 창을 제안하여 사용하였다. 이는 먼저 3×3

창을 4 개의 2×2 창의 영역으로 나누어 각 영역내에서 4 방향(수직, 수평, 대각선, 반대각선)에 대한 gradient operation을 행한 결과가 최소가 되는 영역의 4 화소에 대한 평균을 test 화소의 값으로 두는 방법으로 이는 국부 동일성(local homogeneity)에 바탕을 두고 있다. Operation에 사용된 창들을 그림 1에 나타내었다.

$i-1, j-1$	$i-1, j$	$i-1, j+1$
$i, j-1$	i, j	$i, j+1$
$i+1, j-1$	$i+1, j$	$i+1, j+1$

(a)

-1	-1	-1	1	1	0	-1	-1	-1
1	1	-1	1	-1	1	0	0	1

수평 수직 대각선 반대각선

(b)

그림 1. 가변 평활화를 위한 창들

(a) 원래의 3×3 창

(b) 4 방향 Gradient Operator

Fig. 1. Window for Variable Smoothing.

(a) Original 3×3 Window.

(b) Four Directional Gradient Operators.

$F(i, j)$ 를 화소 (i, j) 에서의 빛의 밝기라 두면 4 영역은 다음과 같이 구성된다.

영역 1 : $R1 = \{(i-1, j-1), (i-1, j), (i, j-1), (i, j)\}$

영역 2 : $R2 = \{(i-1, j), (i-1, j+1), (i, j), (i, j+1)\}$

영역 3 : $R3 = \{(i, j-1), (i, j), (i+1, j-1), (i+1, j)\}$

영역 4 : $R4 = \{(i, j), (i, j+1), (i+1, j), (i+1, j+1)\}$

이렇게 나누어진 각각의 4 영역에 대해 그림 1 (b)에 보여준 4 개의 gradient operator를 적용한다. 여기서는 대각선과 반대각선만을 계산한 다음 이들의 합과 차를 이용하면 수직, 수평성분을 구할 수 있으므로 계산시간을 줄일 수 있다. 즉, 그림 1 (b)에서

수평 = 반 대각선 + 대각선

수직 = 반 대각선 - 대각선

이 된다. 여기서 $SUM(k)$, $k=1, 2, 3, 4$ 를 각 영역에서의 4 가지 방향의 gradient $G(k, 1)$ 의 절대값의

합이라 두면,

$$\text{SUM}(k) = \sum_{i=1}^4 |G(k, i)|, k=1, 2, 3, 4$$

가 된다. 이들 $\text{SUM}(k)$, $k=1, 2, 3, 4$ 중에서 가장 최소값을 구한 다음 이 값을 갖는 영역의 4 화소의 평균값을 평활화한 후의 값으로 둔다. 이 과정을 취함으로써 경계성분을 보존하면서 잡음을 제거할 수 있고 국부적인 동일성도 높일 수 있다. 이의 결과를 그림 5와 6의 (b)에 나타내었다.

2. 임계 분할(thresholding)

임계치 선택 기법에는 원래 영상의 빛의 밝기분포에 따른 histogram을 이용한 전체적인 임계치 선택 기법, gradient operation을 이용한 국부적인 임계치 선택 기법, 그리고 위의 두가지 뿐만 아니라 화소의 위치까지 고려하는 dynamic 임계치 선택 기법 등이 있다.

여기서는 Weszka와 Rosenfeld의 방법을^[1] 텍스춰 영상에 맞도록 개선시킨 것으로 Weszka와 Rosenfeld에 의해 선택된 임계치를 TI이라 하면 텍스춰 화상에서 선택된 임계치 TI과 같은 빛의 밝기를 가지는 화소를 중심으로 하는 3×3 창을 써워 이들의 평균을 임계치로 두었는데 이렇게 함으로써 동일한 영역내에서도, 빛의 밝기 변화가 있는 화상에 대하여도 잘 수행함을 볼 수 있었다. 가변 평활화가 빛의 밝기에 의한 영상의 잡음 제거 및 국부적인 동일성의 증가를 위한 처리라면 임계 분할은 빛의 밝기에 의한 영상을 이진 영상으로 바꾸어 주기 위한 처리 과정이다.

3. 연결된 primitive의 분리

일반적으로 텍스춰 영상을 이진영상으로 임계 분할하게 되면 primitive들을 완전히 분리할 수 없는 경우도 간혹 발생하게 된다. 만약 이들이 한 화소의 크기로 연결되었을 경우에는 그림 2와 같이 3×3 창의 중심화소가 1 값을 갖는 경우에 대해 4 조건 중에서 한 가지 조건이라도 만족하면 그 중심 화소를 0으로 변환하여 준다. 그림에서 X는 don't care 상태를 의미한다. 이와 같은 과정을 수행함으로써 이들을 분리해 낼 수 있을 뿐만 아니라 잡음등으로 인하여 생길 수도 있는

<table border="1"> <tr><td>X</td><td>X</td><td>X</td></tr> <tr><td>0</td><td>1</td><td>0</td></tr> <tr><td>X</td><td>X</td><td>X</td></tr> </table>	X	X	X	0	1	0	X	X	X	<table border="1"> <tr><td>X</td><td>0</td><td>X</td></tr> <tr><td>X</td><td>1</td><td>X</td></tr> <tr><td>X</td><td>0</td><td>X</td></tr> </table>	X	0	X	X	1	X	X	0	X	<table border="1"> <tr><td>X</td><td>X</td><td>0</td></tr> <tr><td>X</td><td>1</td><td>X</td></tr> <tr><td>0</td><td>X</td><td>X</td></tr> </table>	X	X	0	X	1	X	0	X	X	<table border="1"> <tr><td>0</td><td>X</td><td>X</td></tr> <tr><td>X</td><td>1</td><td>X</td></tr> <tr><td>X</td><td>X</td><td>0</td></tr> </table>	0	X	X	X	1	X	X	X	0
X	X	X																																					
0	1	0																																					
X	X	X																																					
X	0	X																																					
X	1	X																																					
X	0	X																																					
X	X	0																																					
X	1	X																																					
0	X	X																																					
0	X	X																																					
X	1	X																																					
X	X	0																																					
수평	수직	대각선	반대각선																																				

그림 2. 연결된 Primitive의 분리를 위한 조건들
Fig. 2. Conditions for Separating Connected Primitives.

작은 점들도 간단히 제거할 수 있으므로 다음의 처리 과정을 위하여 좀 더 깨끗하고 보기 좋은 이진영상 을 얻을 수 있다. 다만 이 과정을 반복해서 수행할 경우 한번의 반복주기 동안 물체의 모서리가 하나씩 없어지게 되므로 반복 회수가 많아질수록 이진 영상은 원래 영상과 틀림 모양을하게 된다. 그러므로 가능하면 적게 반복 수행시키는 것이 좋다.

III. Textural Feature 및 Dominant Primitive 추출

이 장에서는 입력된 텍스춰 영상내에 존재하는 여러 개의 primitive들 중에서 dominant primitive를 추출해내기 위해 전처리 과정을 거친 영상을 이용하여 각각의 primitive들에 대한 모양 해석을 통한 면적, 중심, 둘레, 방향성 등과 같은 textural feature를^[2] 구한 다음 이를 결과에 대한 평균치를 구해서 평균과의 차가 가장 적은 것부터 차례대로 나열하여 전체 primitive수의 20% 내에 들어오는 primitive들만 선택하여 dominant primitive라고 명명한다.

1. Textural feature 추출

여기서는 전처리 과정을 통해 얻은 이진 영상으로부터 각각의 primitive를 구별하기 위해서 서로 연결된 영역들에 label을 붙여 중으로써 영역을 구분한 다음 textural feature를 구한다. 영역들에 대해 번호를 붙여 주는 과정은 다음과 같다.

영상을 주사해 나가면서 가장 먼저 만나는 물체에 label 1을 주고 다음에는 그림 3과 같이 T5가 1인 경우(물체)에 대하여 T1, T2, T3, T4를 사용하여 이들 중 임의의 label을 가지고 있는 것이 있으면 이들 중에서 최소값을 T5에 대입하고 만약 이들이 모두 임의의 label을 가지고 있지 않다면(0인 경우) 다른 영역으로 간주하여 label 값을 하나 증가시켜 새로운 label을 T5에 대입한다.

물체의 형태에 따라 같은 물체임에도 불구하고 다른 영역인 것처럼 labeling을 틀리게 할 수도 있는데 이 경우에는 다시 relabeling이라는 과정을 통하여 틀리게 번호가 대입된 부분을 수정해 주어야 한다. 이는 3×3 창을 사용하여 창내의 9 개 화소값들 중에서 임의의 label을 가지고 있는 것들만을 선택하여 이들 중에서 최소인 값을 T5에 대입해 주는 것으로 전체 영상을 통하여 올바르게 labeling될 때까지 계속해서 반복 수행함으로써 relabeling이 끝나게 된다. 이 과정도 그림 3에 나타내었다. 물론 전 과정은 화소의 값이 1인 것에 대해서만 수행되며 이렇게 각각의 영역에 label을 중으로써 이들 각각은 하나의 독립된 primitive들이 된다.

T 1	T 2	T 3
T 4	T 5	T 6
T 7	T 8	T 9

(a)

0	0	0	0	0	0	0	0	0	0
0	0	0	1	1	1	0	1	1	0
0	1	1	1	1	1	0	1	1	0
0	1	1	0	0	0	0	1	1	0
0	0	0	0	1	1	1	0	0	0

(b)

0	0	0	0	0	0	0	0	0	0
0	0	0	2	2	2	0	3	3	0
0	4	2	2	2	2	0	3	3	0
0	2	2	0	0	0	0	3	3	0
0	0	0	0	5	5	5	0	0	0

(c)

0	0	0	0	0	0	0	0	0	0
0	0	0	2	2	2	0	3	3	0
0	0	0	2	2	2	0	3	3	0
0	2	2	2	2	2	0	3	3	0
0	0	0	0	5	5	5	0	0	0

(d)

그림 3. 연결된 영역의 Labeling 과정

- (a) 번호를 주기 위해 사용하는 3×3 창
- (b) 임계분할에 의한 이진 영상
- (c) Labeling 과정을 한번만 수행했을 경우
- (d) Relabeling에 의해 수정된 후의 결과

Fig. 3. Connected Region Labeling Process.

- (a) 3×3 Window for Labeling.
- (b) Binary Image by Thresholding.
- (c) First labeling.
- (d) Result After Correction by Relabeling.

(1) 면 적

각 영역에 대한 면적은 그 영역내에서 화소의 값이 1인 화소의 수를 전부 합한 것으로 정의한다. 이진 영상이므로 화소의 값이 1인 경우를 $B(i, j)$ 로 나타내고 0인 경우는 $\bar{B}(i, j)$ 로 나타내기로 한다. 면적을 A라고 하면,

$$A = \sum_{(i,j) \in S} \sum_{s} B(i, j)$$

가 된다. 여기서 S는 고려하는 영역에 포함되는 화소들로 이루어진 집합이다.

(2) 중심

각 영역에 대한 중심은 1 값을 갖는 화소의 좌표값 (i 축, j 축)을 전부 합한 다음 이것을 면적으로 나누어 준 것으로 정의한다.^[3] 특히 중심을 구하는데 있어서 디지털 화상의 특성상 각 영역의 폭이 짹수 개로 이루어져 있으면 정수계산만으로는 한 화소 정도가 틀리게 되므로 여기서는 소수점 첫째 자리까지 계산하여 오차를 줄이도록 하였다. 중심의 좌표를 (I, J) 라 두면,

$$I = (1/A) \sum_{(i,j) \in S} \sum_{s} i \times B(i, j)$$

$$J = (1/A) \sum_{(i,j) \in S} \sum_{s} j \times B(i, j)$$

가 된다.

(3) 둘 래

각각의 primitive들은 그들의 모양을 해석하기 위해 경계 성분을 뽑아 내야 한다. 이때 경계 성분의 화소의 수가 둘레가 된다. 경계 성분을 찾는데는 4-이웃화소의 특성을 이용하여 화소의 값이 1인 $B(i, j)$ 에 대해 그의 4-이웃화소인 $(i-1, j)$, $(i, j-1)$, $(i, j+1)$, $(i+1, j)$ 중 하나라도 0인 값을 갖는 경우에 대해서만 (i, j) 를 경계화소로 두어 $Be(i, j)$ 를 1로 둔다. 이를 수식으로 표현하면 다음과 같다.

$$Be(i, j) = \begin{cases} 1, & \bar{B}(i-1, j) \text{ or } \bar{B}(i, j-1) \text{ or} \\ & \bar{B}(i, j+1) \text{ or } \bar{B}(i+1, j) \\ 0, & \text{otherwise} \end{cases}$$

이렇게 하여 얻은 경계 화소는 완전히 한 화소로만 연결된 특성을 가지므로 다음 절에서 수행할 chain-code의 적용에 알맞은 데이터를 제공해 줄 수 있다.

(4) 방향성

Primitive들의 방향을 결정하기 위한 방법에는 여러 가지가 있을 수 있다. 여기서는 그림 4에서와 같은 chain-code를 사용하여 dominant primitive들의 방향을 찾아내는 것으로 (3)에서 얻은 경계 성분에 chain-code를 적용시켜 경계 성분의 방향 변화를 측정한다. 즉, chain-code string 중에 발생빈도가 가장 많은 chain-coded string의 방향을 primitive의 방향으로 결정한다. 물론 여기서 찾아낼 수 있는 방향은 0° , 45° , 90° , 135° 로 제한되지만 대부분의 primitive들은 서로 비슷한 방향으로 위치되어 있기 때문에 이 4 방향이면 primitive를 해석하는데 충분하다. 방향성의 관계를 살펴보면 그림 4에 정의된대로 최대 발생 빈도를 갖는 chain-coded string이 0과 4 일 때에는 0° , 1과 5 일 때에는 45° , 2와 6 일 때에는 90° , 3과 7 일 때에는 135° 방향으로 결정한다. 또한 잡음이나 기타의 요소에 의해서 갑자기 뒤어 나오거나 들어간 부분들과 나지털 화상의 discrete한 특성 때문에 나타나는 몇가지 제한성을 고려해 주어야 한다.^[4]

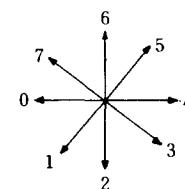


그림 4. Chain-code

Fig. 4. Chain-code.

2. Dominant Primitive 추출

앞에서 네 가지의 texture feature들을 모든 primitive에 대하여 구했다. 여기서는 이들 중에서 판단의 기준으로 적합한 면적, 둘레, 방향성에 대한 평균값을 구해서 이들 평균값과 1 절에서 구한 각각의 영역에 대한 3 가지 feature들(면적, 둘레, 방향성)의 값과의 차이를 구한다. 여기서 방향성은 전체 primitive들이 가지는 방향성 중에 가장 많이 가지는 방향이 평균 방향이 된다. 이렇게 하여 일단 방향성을 먼저 고려하여 평균 방향과 같은 방향을 갖는 primitive들만을 선정한 다음 선택된 것들 중에서 면적 및 둘레의 평균과의 차가 가장 작은 것부터 차례대로 선택하여 전체 primitive 수의 20% 내에 드는 것까지 찾아서 이들을 dominant primitive라고 명명한다.^[3] 즉, A(k)를 각 primitive 들의 면적, P(k)를 primitive들의 둘레, DIR(k, 1), (1=1, 2, 3, 4)를 primitive들의 방향이라고 둔다. 여기서 n을 primitive들의 총수라 할 때 k는 $1 < k < n$ 이며 1은 primitive의 4 방향($1 = 0^\circ, 2 = 45^\circ, 3 = 90^\circ, 4 = 135^\circ$)을 나타낸다. 그리고 면적의 평균을 $A(\text{avg})$, 둘레의 평균을 $P(\text{avg})$ 라 하면,

$$A(\text{avg}) = (1/n) \sum_{k=1}^n A(k)$$

$$P(\text{avg}) = (1/n) \sum_{k=1}^n P(k)$$

이 되며, 각 방향성을 가지는 primitive들의 수를 DIR(1), $1=1, 2, 3, 4$ 이라 하면,

$$\text{DIR}(1) = \sum_{k=1}^n \text{DIR}(k, 1), \quad 1=1, 2, 3, 4$$

이 되고 이들 중 가장 발생 빈도가 많은 방향 성분을 평균 방향 $\text{DIR}(\text{avg})$ 라 두면,

$$\text{DIR}(\text{avg}) = \max(\text{DIR}(1)), \quad 1=1, 2, 3, 4$$

이 된다. 이들을 모두 구한 다음 $\text{DIR}(\text{avg})$ 와 같은 방향을 가지는 primitive들만을 선정하여 평균 면적과 평균 둘레에 대한 차이를 구한다. 면적의 차이를 $A(\text{sub})$ 라 두고 둘레의 차이를 $P(\text{sub})$ 라 두면,

$$A(\text{sub})(k) = |A(k) - A(\text{avg})|, \quad k=1, \dots, m$$

$$P(\text{sub})(k) = |P(k) - P(\text{avg})|, \quad k=1, \dots, m,$$

이 된다. 여기서 m 은 $\text{DIR}(\text{avg})$ 와 같은 방향을 가지는 primitive들의 수이며 최종의 판단 기준은 이 두개의 차이를 합한 값이 된다. 즉,

$$C(k) = A(\text{sub})(k) + P(\text{sub})(k), \quad k=1, \dots, m$$

이 되어 $C(k)$ 가 작은 것부터 나열하여 전체 primitive 수의 20% 내에 드는 primitive들만을 선택하는데 선택된 primitive들을 $\text{DOM}(q)$ 라 하면 $q < n \times 0.2$ 인 판

제가 된다. 이들 선택된 dominant primitive에 대해서 다시 1 절의 과정을 반복 수행하여 이들의 면적, 둘레, 방향성에 대한 평균을 구한 다음, 이 평균과 가장 많은 오직 하나의 dominant primitive를 선택하여 이를 excellent dominant primitive라 명명하며 후에 텍스춰 합성시에 텍스춰 모델을 구하는데 중요한 자료가 된다.

IV. 텍스춰 모델 및 공간 구조 추출

여기서는 III장에서 구한 excellent dominant primitive와 각각의 primitive들에 대한 중심좌표를 이용하여 텍스춰 모델 및 primitive들 사이의 공간적인 배열 구조를 해석한다. 물론 텍스춰 모델은 영상내에 존재하는 모든 primitive들의 표준이 되며 두 vector를 이용하여 이 모델을 이차원적으로 반복해서 나열할 수 있다.

1. 텍스춰 모델 추출

이 과정은 아주 간단한 것으로 III장에서 구한 excellent dominant primitive의 위치, 면적, 모양 등을 파악한 다음 이와 일치하는 원래 영상의 위치 및 그 위치에서의 빛의 밝기에 해당되는 값을 기억 장소에 기억시킨다. 텍스춰 모델 및 구조해석을 정확히 하기 위해서는 무엇보다도 이진 영상으로 변환해 주는 임계분할이 잘 되어야 한다.

2. 제안한 공간 구조 추출 방법

이 과정이 본 연구에서 매우 중요한 부분이다. 왜냐하면 primitive들 사이에서 반복적으로 발생하는 거리 vector를 정확히 찾아내지 못하면 다음 과정인 텍스춰 합성에서 원래 영상과는 다른 영상을 만들어 내기 때문이다.

[3]에서는 전체 primitive들 중에서 15~20% 내에 해당하는 dominant primitive들의 중심 좌표만을 고려하는데 이의 단점은 만약 dominant primitive들이 서로 분산되어서 선택되는 경우에는 원래의 배열과는 전혀 다른 거리 vector를 유도할 수도 있다는 것이다. [5]에 따르면 모든 primitive들의 중심좌표를 전부 고려하여 이들 각 중심 좌표와 다른 중심들과의 거리 vector를 모두 구해야 하므로 계산 시간이 너무 많이 소요되는 단점을 가지고 있다. 그러므로 여기서는 계산 시간도 상당히 줄이며 계산 과정도 간단한 방법을 제안하였다. 즉, III장에서 구한 excellent dominant primitive로부터 가장 가까이 위치하는 8 개의 primitive들의 중심좌표를 구한 다음 이들 각 primitive들의 중심좌표와 excellent dominant primitive의 중심좌표와의 거리를 계산한 다음 그 값을 작은 것부터 순서대로 나열한다. 이렇게 한 다음 최소이면서 서로 다른 두 개의 거리

vector를 선택하면 이것이 바로 primitive들 사이의 공간 구조를 표시해 주는 regularity vector가 된다. 여기서 가장 중요한 것은 excellent dominant primitive와 가장 가까운 8개에 속한다고 할지라도 그보다 앞에서 선택된 거리 vector와 같은 tangent 성분을 가지면 지금의 것은 8개의 거리 vector에 포함시키지 않음으로써 같은 방향을 갖는 거리 vector의 선택을 방지할 수 있다.

V. 텍스춰 합성

IV장에서는 텍스춰 모델과 그들 사이의 공간적인 배열 구조를 규명해 주는 regularity vector를 구했다. 여기서는 이를 두 가지를 이용하여 텍스춰 영상을 합성해 낸다. 영상은 이 차원 평면에 나타나는 것이므로 두 개의 vector의 합과 차에 의해서 완전히 묘사될 수 있다. 그래서 본 논문에서는 하나의 텍스춰 모델과 두 개의 regularity vector를 이용하여 전체 영상을 합성한다. 먼저 전체 영상을 평균 background gray-level로 채운다. 여기서 평균 background gray-level은 텍스춰 영상을 이진 영상으로 임계 분할할 때 primitive가 아닌 부분의 빛의 밝기에 대한 평균을 구하여 얻는다. 그런 다음 텍스춰 모델을 두 개의 regularity vector에 의해 반복적으로 배열해 나가면 결국 전체 영상은 똑같은 텍스춰 모델이 regularity vector의 크기 만큼 떨어진 곳에 위치하게 된다. 합성시에 영상 frame 내부는 별 문제가 없지만 영상 frame의 경계영역에서는 regularity vector를 계속 더하다 보면 frame을 벗어나는 경우 모델 fitting을 하지 못하는 부분이 발생하는데 이를 방지하기 위해 frame 크기에 regularity vector 크기의 반을 더한 것 만큼을 합성 영역으로 삽아준다. 이렇게 하면 영상 frame의 경계 부근의 모델 fitting도 가능하다.

VI. 실험 및 검토

본 논문의 실험을 위하여 사용된 컴퓨터는 IBM-PC/XT이고 사용 언어는 assembly 언어이며 실험에 사용된 영상은 128×128 의 크기로 256 level의 빛의 밝기를 갖는다. 그림 5는 인공적으로 합성한 SYNTH1 영상에 대한 실험 결과이고 그림 6은 Brodatz Album^[1]에 있는 D22 텍스춰 영상에 대한 실험 결과를 보여 준다. 그림에서 검은 부분은 해석의 초점이 되는 반복 배열 구조를 갖는 텍스춰의 primitive들이고 흰 부분은 텍스춰의 background에 해당되는 부분이다. 그림 5와 6에서 볼 수 있듯이 이진 영상의 해석 결과 얻어진 합성 텍스춰 영상은 실제와 비슷하다. 다만 background의 빛의 밝기가 균일하므로 실제 영상에 비해 다소 단조

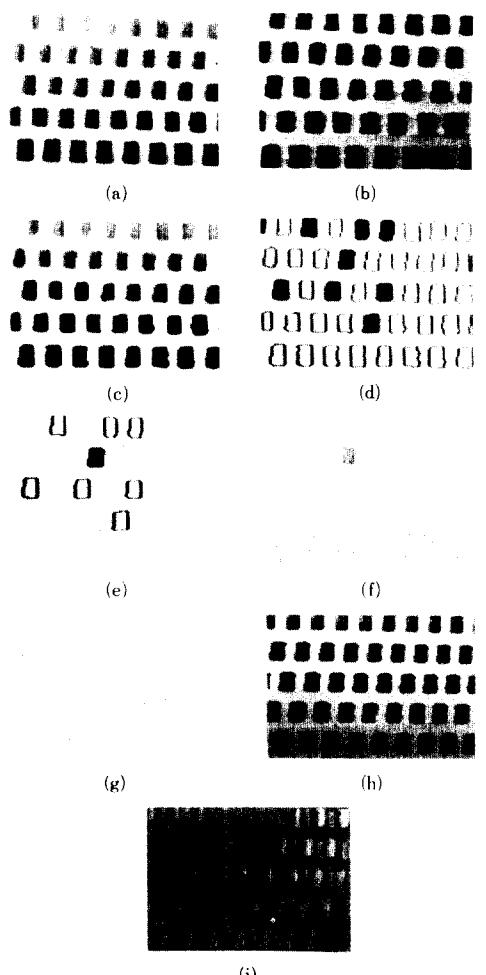


그림 5. SYNTH1 영상에 대한 해석 결과

- (a) 원래의 SYNTH1 영상
- (b) 가변 평활화 후의 영상
- (c) 전처리 과정을 거친 이진 영상
- (d) Dominant Primitive들
- (e) Excellent Dominant Primitive
- (f) 텍스춰 모델
- (g) Regularity Vector
- (h) 합성한 텍스춰 영상
- (i) (a)와 (h)의 차영상

Fig. 5. The Results of Texture Analysis for a SYNTH1 Image.

- (a) Original SYNTH1 Image.
- (b) Image After Variable Smoothing.
- (c) Binary Image After Preprocessing.
- (d) Dominant Primitives.
- (e) Excellent Dominant Primitive.
- (f) Texture Model.
- (g) Regularity Vectors.
- (h) Synthesized Texture Image.
- (i) Difference Image Between (a) and (h).

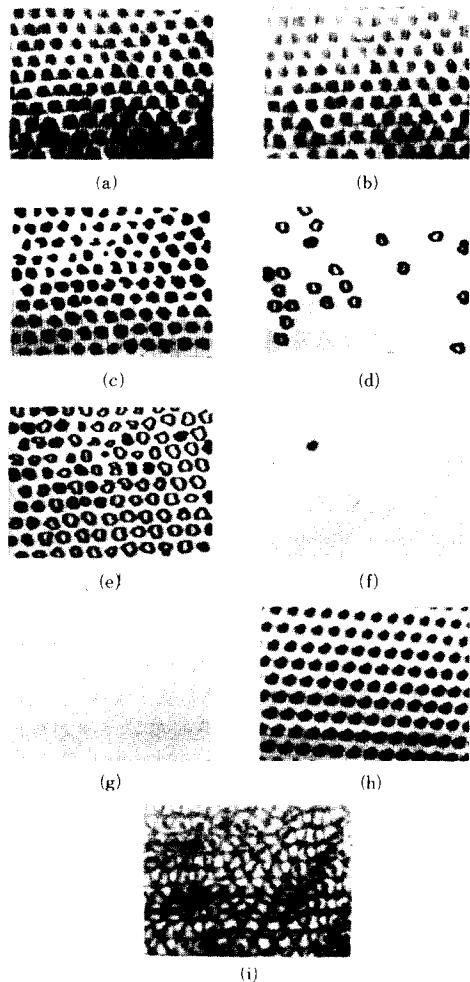


그림 6. D22 영상에 대한 해석 결과

- (a) 원래의 D22 영상
- (b) 가변 평활화 후의 영상
- (c) 전처리 과정을 거친 이진 영상
- (d) Dominant Primitive들
- (e) Excellent Dominant Primitive
- (f) 텍스처 모델
- (g) Regularity Vector
- (h) 합성한 텍스처 영상
- (i) (a)와 (h)의 차영상

Fig. 6. The Results of Texture Analysis for a D22 Image.

- (a) Original D22 Image.
- (b) Image After Variable Smoothing.
- (c) Binary Image After Preprocessing.
- (d) Dominant Primitives.
- (e) Excellent Dominant Primitive.
- (f) Texture Model.
- (g) Regularity Vectors.
- (h) Synthesized Texture Image.
- (i) Difference Image Between (a) and (h).

로운 느낌을 받을 수 있다. 오차는 excellent dominant primitive에서 멀어질수록 커지는데 이는 거리 vector를 구할 때 excellent dominant primitive 주위의 8개의 primitive들 만을 고려했기 때문이다. 그렇지만 오차의 크기는 그렇게 크지 않음을 눈으로도 볼 수 있다. 표 1, 2는 III-1절에서 설명한 것처럼 SYNTH1 영상에 대한 primitive들의 textural feature와 dominant primitive들에 대한 textural feature를 나타내며 표 3은 실험 텍스처 영상에 대한 regularity vector를 표시한

표 1. SYNTH1에 대한 Primitive들의 Textural Feature

Table 1. Textural Features of Primitives for a SYNTH1 image.

maximum brightness: 0066

region	brightness	area	center	perimeter	direction
0001	0015	0032	0066.8, 0071.8	0025	0090 d
0002	0016	0117	0078.3, 0073.0	0038	0090 d
0003	0017	0134	0092.3, 0073.9	0042	0090 d
0004	0018	0115	0107.4, 0074.6	0038	0090 d
0005	0020	0127	0122.3, 0075.9	0040	0090 d
0006	0021	0132	0137.2, 0076.9	0041	0090 d
0007	0022	0164	0152.6, 0077.7	0046	0090 d
0008	0023	0155	0168.2, 0077.6	0043	0090 d
0009	0024	0153	0184.2, 0078.7	0043	0090 d
0010	0026	0112	0070.7, 0097.2	0037	0045 d
0011	0028	0116	0084.5, 0098.0	0038	0090 d
0012	0029	0131	0098.8, 0098.7	0040	0045 d
0013	0031	0134	0113.3, 0099.6	0041	0090 d
0014	0032	0141	0128.4, 0100.9	0042	0090 d
0015	0033	0146	0144.3, 0101.1	0042	0090 d
0016	0034	0153	0159.9, 0102.2	0043	0090 d
0017	0035	0154	0175.3, 0102.4	0044	0090 d
0018	0036	0045	0187.9, 0103.3	0031	0090 d
0019	0037	0138	0076.2, 0123.0	0042	0090 d
0020	0038	0120	0090.5, 0123.2	0039	0090 d
0021	0039	0136	0105.1, 0124.6	0041	0090 d
0022	0040	0123	0120.2, 0125.6	0039	0045 d
0023	0042	0133	0134.9, 0126.0	0040	0090 d
0024	0043	0147	0150.8, 0127.0	0043	0045 d
0025	0045	0161	0166.6, 0128.1	0045	0090 d
0026	0046	0165	0183.6, 0128.7	0045	0045 d
0027	0048	0090	0068.5, 0147.6	0035	0090 d
0028	0049	0122	0082.4, 0148.9	0039	0090 d
0029	0051	0140	0097.0, 0149.5	0042	0090 d
0030	0052	0124	0111.7, 0150.1	0041	0090 d
0031	0053	0139	0126.9, 0150.8	0042	0090 d
0032	0055	0143	0142.1, 0151.5	0042	0090 d
0033	0056	0140	0159.0, 0152.5	0042	0090 d
0034	0057	0173	0174.4, 0153.0	0046	0090 d
0035	0058	0055	0187.7, 0153.4	0034	0090 d
0036	0059	0144	0074.3, 0174.7	0043	0045 d
0037	0060	0151	0089.0, 0175.2	0044	0090 d
0038	0061	0141	0103.7, 0176.0	0044	0090 d
0039	0062	0157	0118.9, 0176.5	0046	0090 d
0040	0063	0140	0134.2, 0177.2	0043	0090 d
0041	0064	0155	0150.4, 0177.8	0045	0090 d
0042	0065	0174	0165.6, 0178.5	0046	0090 d
0043	0066	0178	0183.2, 0179.3	0047	0090 d
average values		: 0133		0041	0090 d
Primitive center		: 0137.2, 0076.9			
Dominant primitive		: 0021			

표 2. SYNTH1 영상에 대한 Dominant Primitive 들의 Textural Features

Table 2. Textural Features of Dominant Primitives for a SYNTH1 Image.

maximum brightness : 0053

region	brightness	area	center	perimeter	direction
0001	0017	0134	0092.3, 0073.9	0042	0090 d
0002	0020	0127	0122.3, 0075.9	0040	0090 d
0003	0021	0132	0137.2, 0076.9	0041	0090 d
0004	0031	0134	0113.3, 0099.6	0041	0090 d
0005	0037	0138	0076.2, 0123.0	0042	0090 d
0006	0039	0136	0105.1, 0124.6	0041	0090 d
0007	0042	0133	0134.9, 0126.0	0040	0090 d
0008	0053	0139	0126.9, 0150.8	0042	0090 d
average values		0134		0041	0090 d
Primitive center		0113.3, 0099.6			
Dominant primitive		0031			

표 3. Regularity Vector

Table 3. Regularity Vectors.

SYNTH1 영상 :
Regularity vector-1 : 0014, 0001
Regularity vector-2 : 0005, 0025
D22 영상 :
Regularity vector-1 : 0009, 0001
Regularity vector-2 : 0004, 0013

것이다. 그리고 표 4는 계산에 필요한 연산 수를 기준의 방법들과 비교한 것인데 표에서도 알 수 있듯이 기준의 알고리즘들은 regularity vector를 구하기 위해 모든 primitive의 중심 좌표를 이용하는지^[1] 혹은 전체 primitive 수의 20%를 이용하는데^[2] 전자에서는 너무 많은 계산량을 요구하며 후자에서는 올바른 vector를 구할 수 없는 단점을 가진다. 본 논문에서는 계산량도 줄이고 vector도 정확하게 구할 수 있도록 excellent dominant primitive와 가장 인접한 8 개 primitive들의 중심좌표만을 고려하였으며 기준의 알고리즘들은 primitive 수에 따라 계산량이 증가하지만 제안한 방법은 primitive 수에 무관하게 계산량이 일정한 장점을 갖는다.

표 5는 해석 결과 얻은 영상 정보 압축 효과를 보여주는데 일반적으로 텍스처 영상은 그들을 구성하고 있는 primitive들이 서로 유사한 특성을 갖는 경우에는 그들의 모양 해석과 구조 해석을 통하여 여러 개의 primitive들 중에서 가장 풍통성이 많은 하나의 primitive와 그들의 반복 구조를 결정해 주는 두개의 regularity vector를 구함으로써 극히 적은 정보만으로도

표 4. Regularity Vector를 구하는데 필요한 연산 수

Table 4. The Number of Computation for Calculating Regularity Vectors.

제한자	[1]	[3]	본 논문
계산할 vector의 수 (n은 primitive의 수) (m = n × 2)	nC ₂	mC ₂	8
하나의 vector 계산에 들어가는 총 연산수	6	6	6
전체 연산 수	3n(n - 1)	3m(m - 1)	48
비교 (연산수의 변화)	n에 따라 변함	m에 따라 변함	정

표 5. 정보 압축 효과

Table 5. Data Compression Effects.

	영상 전체를 전송	regularity vector 를 이용할 경우	각 primitive의 중심을 이용할 경우
image frame	128 × 128	SYNTH1	D22
gray-level(bits)	8	8	8
texture model size		134	46
regularity vector		2 × 7	2 × 7
background(bits)		8	8
중심 좌표			43 × 7
전체 bit 수	128 × 128 × 8	134 × 8 + 2 × 7 + 8	46 × 8 + 2 × 7 + 8
백분비(%)	100	0.8	0.3
		1.1	0.9

텍스처 영상 전체를 인식하고 표현할 수 있는 장점을 얻을 수 있다는 점을 확인할 수 있었다.

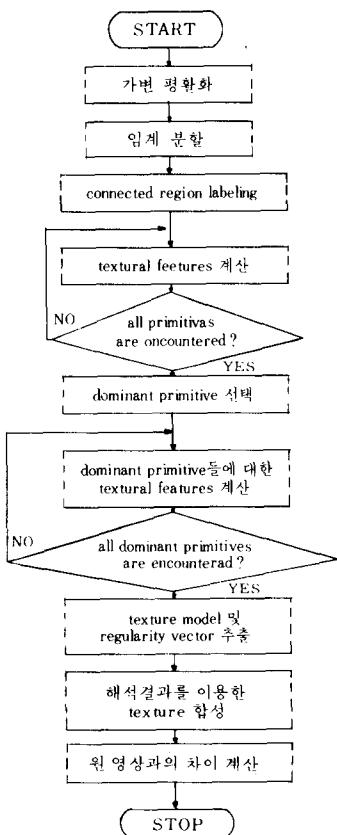
본 논문에서 고려한 것은 구조적으로 반복적인 특성을 갖는 텍스처 영상의 해석인데 좀 더 고려할 수 있는 것은 같은 형태의 primitive들로 구성되어 있으나 그들 사이의 반복 구조는 존재하지 않는 random 텍스처 영상의 효과적인 묘사이다. 이는 regularity vector의 계산이 필요하지 않기 때문에 계산 시간도 줄일 수 있을 뿐만 아니라 어떤 구조를 갖더라도 이들을 적은 양의 정보로 묘사할 수 있는 장점을 가진다. 즉, 텍스처 모델의 설정은 앞의 IV장과 같으나 regularity vector는 계산하지 않고 다만 textural feature를 구할 때 계산한 각 primitive들의 중심좌표만 주어지면 이들과 텍스처 모델의 중심과의 상대적인 위치 계산에 의

해 각 primitive들의 위치에 텍스처 모델이 채워지므로 고려의 대상이 되는 모든 primitive들을 묘사할 수 있게 된다.

조금의 오차를 감수할 경우 정보 전송에 있어서 보내 주어야 할 정보의 양은 텍스처 모델과 두개의 regularity vector 및 background에 대한 빛의 밝기 정보이므로 예로 든 두 영상을 비교해 볼 때 실제의 영상을 모두 전송할 경우에 비하여 약 100:1로 정보량을 압축할 수 있으며(표 5 참조) 이런 구조를 갖는 텍스처 영상의 segmentation 및 합성도 가능하게 된다. 본 논문에서 제안하고 적용한 알고리즘에 대한 flow chart를 표 6에 나타내었다.

표 6. 전체 시스템의 흐름도

Table 6. The Flow Chart of Overall System.



VII. 결 론

구조적으로 반복적인 배열을 갖는 텍스처 영상의 해석에서 가장 중요한 것은 짧은 시간에 텍스처 모델과 공간 구조를 구하는 것인데 본 논문에서는 regularity vector를 구하는데 있어서 기존의 방법들에 비해 시간

을 단축시켰으며 계산과정도 간단하게 하였다. 그리고 텍스처 영상에 적용할 수 있는 몇몇의 개선된 전처리 알고리즘들을 제안하여 사용하였다. 영상이 구조적인 특성을 갖는 경우에는 고 압축 데이터 전송이 가능하며 이들의 성질을 이용하여 구조적 특성을 갖는 텍스처 영상의 분할도 수행할 수 있다.

본 논문에서는 간단한 이차원 구조만을 고려했었는데 좀 더 확장하여 여러 형태의 구조, 즉 계층 구조를 갖는 영상에 대한 연구가 더욱 필요하며 나이가 동일한 모양을 갖는 것 뿐만 아니라 여러 개의 서로 다른 모양을 갖는 primitive들이 같이 존재하는 경우도 고려해 봄직 하다.

参 考 文 献

- [1] F. Tomita, Y. Shirai, and S. Tsuji, "Description of textures by a structural analysis," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. PAMI-4, no. 2, pp. 183-191, March 1982.
- [2] R.M. Haralick, "Statistical and structural approaches to textures," *Proc. IEEE*, vol. 67, no. 5, pp. 786-804, 1979.
- [3] J.G. Leu and W.G. Wee, "Detecting the spatial structure of natural textures based on shape analysis," *Comput. Vision, Graphics, Image Processing*, vol. 31, pp. 67-88, 1985.
- [4] F.M. Vilnrotter, R. Nevatia, and K.E. Price, "Structural analysis of natural textures," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. PAMI-8, no. 1, pp. 79-89, Jan. 1986.
- [5] T. Matsuyama, K. Saburi, and M. Nagao, "A structural analyzer for regularly arranged textures," *Comput. Graphics, Image Processing*, vol. 18, pp. 259-278, 1982.
- [6] F. Tomita and S. Tsuji, "Extraction of multiple regions by smoothing in selected neighborhood," *IEEE Trans. Syst. Man. Cybern.*, vol. SMC-7, no. 7, pp. 107-109, Feb. 1977.
- [7] J.S. Weszka and A. Rosenfeld, "Histogram modification for threshold selection," *IEEE Trans. Syst. Man., Cybern.*, vol. SMC-9, no. 1, pp. 38-52, Jan. 1979.
- [8] K.S. Fu, *Syntactic Pattern Recognition and Applications*. Prentice-Hall, pp. 79-128, 1977.
- [9] P. Brodatz, *Textures*. Dover, 1966. *