

## 방향을 고려한 효율적이고 디테일한 텍스처 합성

추연희<sup>o</sup>, 김종현<sup>\*</sup>

<sup>o</sup>강남대학교 소프트웨어융합학부,

<sup>\*</sup>인하대학교 소프트웨어융합대학(디자인테크놀로지학과)

e-mail: jonghyunkim@inha.ac.kr<sup>\*</sup>

## Efficient and Detailed Texture Synthesis with Orientation Considerations

Yeon Hee Choo<sup>o</sup>, Jong-Hyun Kim<sup>\*</sup>

<sup>o</sup>School of Software Application, Kangnam University,

<sup>\*</sup>College of Software and Convergence (Dept. of Design Technology), Inha University

### ● 요약 ●

본 논문에서는 텍스처 합성할 때 방향을 고려하여 합성의 품질의 개선시킬 수 있는 방법을 제안한다. 또한 고정된 회전 각도가 아닌, 다양한 각도를 자동으로 샘플링하여 효율적으로 예제 이미지를 생성할 수 있도록 하였고, 이를 통해 합성 경계간의 차이를 자연스럽게 완화시킬 수 있는 결과를 실험을 통해 보여준다.

**키워드:** 텍스처 합성(Texture synthesis), 회전 텍스처 이미지(Rotated texture image), 방향성(Orientation)

### I. Introduction

본 논문에서는 하나의 텍스처 이미지로부터 다양한 회전 값이 적용되어 보다 자연스럽게 원본 이미지보다 큰 크기의 텍스처를 합성하는 것을 목표로 한다. Park *et al.*에서 소개되었던 다수의 회전 예제 이미지를 이용한 알고리즘을 응용하여 기존 퀴팅(Quilting) 합성으로 반복적인 패턴과 경계선 생성 문제를 개선하였다[1,2]. 주어진 텍스처 이미지의 크기를 키우기 위해서는 텍스처 이미지 크기 밖의 나머지 부분을 만들어내는 과정이 필요하며, 이때 사용되는 방법이 텍스처 이미지 합성이다. 기존 기법에서는 패치 블록(Patch block)을 겹치게 이어붙이는 합성 방법을 사용하였다[2]. 먼저 랜덤하게 하나의 블록을 가져온 후 그다음에 패치 배치될 블록부터는 이전의 블록과 겹치는 영역의 오차를 계산하여 오차가 낮은 값들을 가져온다. 이어붙일 때는 겹치는 영역에서 최소비용경로(Minimum cost path)를 계산 자연스럽게 합성된 이미지를 만든다. 하지만 기존의 이미지 합성 기법은 겹치는 부분에서의 오차가 가장 작은 패치가 주어진 텍스처에서는 연관성이 떨어질 수 있다. Park *et al.*은 이 문제를 보완하기 위해 다수의 회전 예제 이미지를 제시하여 경계선 생성과 반복적인 패턴을 해결하는 알고리즘을 제시하였다[1].

### II. The Proposed Scheme

#### 1. Preprocessing

결과물에 다양한 회전 값을 적용하기 위해 주어진 입력 텍스처 이미지를 중심점을 기준으로  $\theta$ 만큼 회전시켜  $N$ 개의 회전 텍스처 이미지를 준비한다. Park *et al.*에서 소개되었던 알고리즘에서는 이미지 회전 시 이미지가 존재하지 않는 부분을  $\theta$ 가 90의 배수이면 타일링, 90의 배수가 아니면 미러링을 통해 채웠다. 하지만 이런 방법은 채우는 과정에서 경계선이 생기기 때문에 이렇게 생겨난  $N$ 개 회전 텍스처 이미지에서 패치 블록의 겹치는 영역의 오차만을 고려하여 다음 패치 블록의 후보를 가져오면, 패치 블록에서 겹치는 영역 외의 부분에서 경계선이 존재하는 채로 합성될 수 있어 최종 결과물에 다수의 경계선이 나타나 자연스러운 결과물이 나오지 않는 문제가 발생한다. 본 논문에서는 마스크를 만들 때 이미지가 존재하지 않는 부분의 영역을 계산하기 위해 이미지의 높이가 너비와 같다고 가정했으며,  $\theta$ 는 다음과 같이 설정했다 ( $\theta = 45, 135, 225, 315$ ). 이 과정에서 존재하지 않는 영역은 Fig. 1에서 보듯이 2변의 길이가 같은 직각 삼각형의 형태가 나오는 것을 확인할 수 있다.

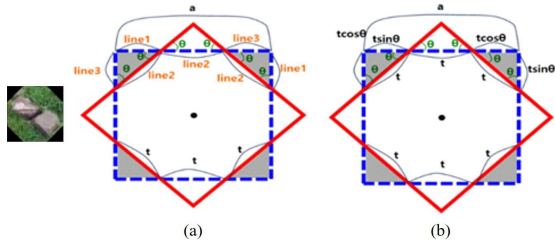


Fig. 1. Example of what is and isn't present when rotating an image.

이때 직각삼각형의 크기 4개는 다 동일하다. 이미지가 존재하지 않은 영역을 왼쪽 상단부터 반시계방향으로  $Z_{1...4}$ 라고 가정한다. 본 논문에서는 Fig. 1 같이  $line_{1...3}$ 를 계산한다 (수식 1~3 참조).

$$line_1 = t \sin \theta \tag{1}$$

$$line_2 = t \tag{2}$$

$$line_3 = t \cos \theta \tag{3}$$

여기서  $a$ 와  $t$ 는 다음과 같이 계산한다 (수식 4~5 참조).  $Z_{1...4}$ 를 두 변의 길이가  $line_1$ 인 직각 이등변 삼각형으로 간주하고 영역에 대한 마스크를 설정했다.

$$a = t \sin \theta + t + t \cos \theta \tag{4}$$

$$t = \frac{a}{\sin \theta + 1 + \cos \theta} \tag{5}$$

이미지가 회전되면서 입력 텍스처 이미지가 차지하고 있던 공간에서 존재하지 않는 빈 곳이 생기게 된다. 합성 시 이미지가 존재하는 공간에서만 참고하기 위해 마스크를 사용하여 구분하였다. 합성에서의 좌표는 입력 텍스처 이미지의 색상인  $c$ 를 포함해서 3채널로 표현 ( $x, y, c$ )했으나, 회전 텍스처 이미지는 회전까지 추가하여 4채널 ( $x, y, c, r$ )로 표현했다. 본 논문에서는 회전된 텍스처 이미지에서 존재하는 공간과 아닌 공간을 마스크하기 위해 입력 텍스처 이미지는 높이  $h$ 와 너비  $w$ 의 길이가 같도록 설정했다.

## 2. Texture Synthesis

합성을 시작할 때는 주어진 입력 텍스처 이미지에서 랜덤하게 패치 블록을 하나 가져온다. 이때, 패치 블록의 크기는 사용자가 임의로 정한다. 패치 블록과 겹치는 크기는 이웃과 겹치는 부분이 패치 블록의 특징을 충분히 잘 반영할 만큼 적당해야 한다. 본 논문에서는 패치 블록의 크기는  $20 \times 20$ 으로, 겹치는 부분은 패치 블록의  $\frac{1}{6}$ 로 설정했다. 회전 텍스처에서 이미지 값이 존재하는 공간의 모든 위치에서의 패치 블록에 대한 오차를 계산하고, 오차가 작은 것부터 5개의 패치 블록을 선별한다. 오차가 임계값 이하일 경우의 패치 블록들을 찾는 방식도 있지만, 선택된 패치 블록들이 하나밖에 없는 상황이 발생할 수도 있기 때문에 오차가 작은 순으로 가져오는 방식을 이용하였다. 가장 작은 오차를 가지는 패치 블록만을 사용하면 같은

패치 블록이 자주 발생하여 반복적이고 인위적인 결과가 나올 수 있으므로, 합성 시 다양성을 주기 위하여 회전된 후보군 패치 블록에서 랜덤하게 패치들을 가져온다. 이전 패치 블록과 새로 가져온 패치 블록 간의 겹치는 부분에서의 최소비용경로를 계산하여 새로 가져온 패치 블록의 경계선으로 계산하고, 이들을 이어 붙인다. Fig. 2는 우리의 방법과 이전 기법을 비교한 결과이다. 회전을 고려한 방식이기 때문에 이전 기법들보다 자연스러운 결과를 얻을 수 있었다.

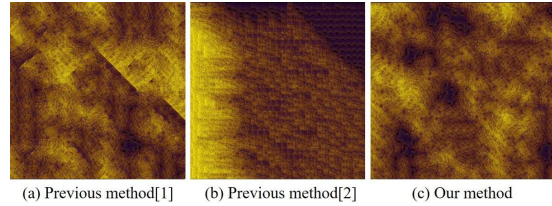


Fig. 2. Comparison results with our method and previous method.

## III. Conclusions

본 논문에서는 회전 텍스처를 통해 방향성을 고려한 합성 결과를 생성하여 기존의 텍스처 합성 기법을 개선했다. 본 논문에서는 제한된 회전이미지를 사용하였기 때문에 참고할 수 있는 이미지의 개수가 늘었다고 해도 자연스러운 합성을 하지는 못하는 한계가 있다. 패치 블록으로 합성 시 겹치는 오차만 고려하므로 패치 블록에서 겹치는 부분 이외에 영역에서 경계가 패치 블록을 가져올 경우 부자연스러운 합성 결과가 생성된다는 문제가 존재한다. 향후 이 문제를 해결할 계획이다.

## REFERENCES

[1] Han-Wook Park, Chang-Hun Kim, "Parallel Rotated Exemplar-based Texture Synthesis", Journal of the Korea Computer Graphics Society, Vol. 15, No. 1, pp. 17-23, 2009.

[2] Efros, Alexei A., and William T. Freeman. "Image quilting for texture synthesis and transfer." In Proceedings of the 28th annual conference on Computer graphics and interactive techniques, pp. 341-346. 2001.