



Video based Cartoon Rendering using Back-map and Edge-map

Jae-Khun Chang, Hyeon-Jin Park, Seung-Taek Ryoo*

School of Computer Engineering, HanShin University

ABSTRACT

The purpose of cartoon style rendering is to produce traditional cartoon stylized look such as comic books and cartoon movies. The essential of cartoon rendering is the black outline (inking) and the solid color of the interior regions (painting). In this paper, we suggest video based cartoon rendering using temporal frame difference. Video based cartoon rendering has problems occurred by temporal coherence. One of these problems is difficult to find accurate ink-lines from moving objects. Also, waving effect in the boundary area of painting is appeared by illumination variation. To resolve waving effect, we use back-map applied the difference between the current frame and the reference frame in painting step. Corrected frame can be generated from blending reference frame and current frame using back-map. Corrected frame is transformed into color based segmented frame using the mean shift algorithm. To improve inaccurate boundary edge, we use edge-map obtained by adding current edge extracted frame to edge difference between previous and current edge extracted frames in inking step. Blurred frame was generated by bilateral algorithm from color segmented frame. To extract boundary edge, we use sobel edge detection algorithm. The result image can be produced by subtracting edge-map from color segmented frame.

© 2016 KKITS All rights reserved

KEYWORDS : Non-photorealistic rendering, Stylized rendering, Cartoon rendering, Temporal coherence, Realtime video painting

ARTICLE INFO: Received 7 November 2016, Revised 12 December 2016, Accepted 12 December 2016.

1. 서론

*Corresponding author is with the School of Computer Engineering, HanShin University, 137 Hanshin Univ.-gil Osan-si, Gyeonggi-do, 18101, KOREA.
E-mail address: stryoo@hs.ac.kr

스마트폰이 발전함에 따라 스마트폰의 카메라 어플리케이션 또한 계속해서 발전 중이며 요즘은

포토샵이 아닌 스마트폰으로도 사진 보정 작업이 가능해졌다. 그 보정 작업 중 비사실적 렌더링(Non-Photorealistic Rendering : NPR)을 적용하는 필터들이 많아지면서 비사실적 렌더링의 관심도 높아졌다. 그러나 대부분 정지 영상에 적용하는 기술이기 때문에 비디오 영상에 비사실적 렌더링을 적용하고자 한다.

비사실적 렌더링은 사람이 직접 그리거나 만들어낸 것과 같은 예술적인 영상들을 컴퓨터로 재현해 내어 인간의 감성에 자극을 줄 수 있는 콘텐츠를 제작하는 것이 목적이며 1990년대에 들어 컴퓨터그래픽스 분야의 중요한 연구 주제 중의 하나가 되었다[1]. 본 논문에서는 비사실적 렌더링 중에서 카툰 렌더링(Cartoon Rendering)을 고정된 이미지가 아닌 비디오 영상에 적용한다. 카툰 렌더링을 고정된 이미지가 아닌 비디오 영상에 적용할 경우 일반적으로 빛에 영향을 받는 색상의 경계 부분이 일렁이게 된다. 본 논문에서는 이러한 문제를 해결하기 위해 배경맵을 사용한다. 또한 고정된 이미지가 아닌 실시간 영상에 선화(Inking) 작업을 하게 되면 움직이는 물체에 대한 선화 결과가 정확하지 않게 나오는데 이러한 문제점을 해결하기 위해 본 논문에서는 엣지맵을 이용한다.

2. 관련연구

카툰 렌더링은 만화나 코믹 서적에서 볼 수 있는 그림과 같이 사실적이지 않은 스타일로 개체를 렌더링하는 것을 말한다. 카툰 렌더링은 크게 선화(Inking)와 채색(Painting) 2가지 작업으로 나뉘어진다. 다양한 카툰 연구 중 초기 연구[2]를 시작으로 90년대 후반부터 실제 사람의 표현방식을 컴퓨터 그래픽스 기술로 재현하는데 초점을 맞춘 연구와 3차원 모델을 이용하여 2차원 셀 애니메이션과 같은 만화적인 스타일의 만화 영상을 제작하기 위한

셰이더 및 플러그인들이 개발 되었다[3,4]. 초기에는 톤 셰이딩 방법에 의해 만화와 같은 단순한 채색 효과와 채색에 따른 영상의 느낌에 대한 연구 그리고 외곽선 검출을 위한 여러 가지 기법들이 개발 되었다[4,5]. 또한, 화면의 확대나 축소시 시간의 진행에 따라 발생하는 영상의 끊김 현상을 해결하기 위해 애니메이션 일관성 유지를 위한 연구가 진행 되었다[6]. 이후에는 고급 톤 렌더링 기술 [7-9] 및 이러한 톤 렌더링 영상을 실시간으로 렌더링 하여 사용자와의 상호작용이 가능하도록 하는 연구[10,11]가 진행되었다.

비디오 기반 카툰 렌더링 기술을 개발하기 위해서는 공간적/ 시간적인 일관성 유지 문제가 대두된다. 본 논문에서는 기존 카툰 렌더링 방법에서 발생한 일관성 유지 문제를 해결하기 위해 배경맵과 엣지맵을 이용한 실시간 카툰 렌더링 방법을 제안한다.

3. 비디오 기반 카툰 렌더링

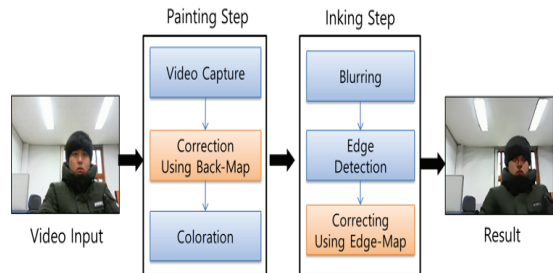


그림 1. 제안된 비디오 기반 카툰렌더링의 구성
Figure 1. The overview of suggested video based cartoon rendering

<그림 1>은 배경맵과 엣지맵을 이용한 카툰 렌더링 시스템의 간략한 흐름도를 보여주고 있다. 채색 단계에서는 먼저 캠을 통해 영상을 받아오고 배경맵을 이용한 수정 단계에서 배경맵을 이용하여 영상을 보정한다. 보정된 영상을 선화 단계에서

평균 이동 알고리즘을 이용하여 영상을 색상 기반으로 분할한다. 다음 선화 단계에서는 채색 단계에서 분할한 영상을 블러링 단계에서 양방향 알고리즘을 이용하여 블러링을 해주고 외곽선 검출 단계에서 외곽선 검출 방법의 하나인 소벨 외곽선 검출 방법을 사용하여 외곽선을 추출한다. 소벨 외곽선 검출을 통해 추출된 영상은 엣지맵 수정단계에서 엣지맵을 이용하여 영상을 보정해 주고 결과 영상을 출력한다. 각 단계의 상세한 설명은 다음 장에서 설명할 것이다.

3.1 채색

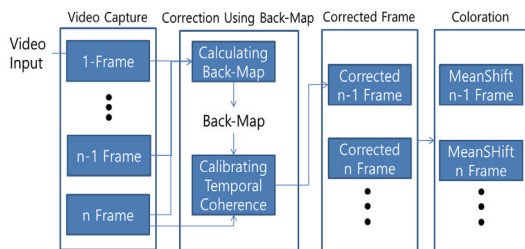


그림 2. 채색 단계의 시스템 흐름도
Figure 2. System flowchart of painting step

<그림 2>는 채색 단계의 상세한 시스템 흐름도이다. 영상 캡처 단계에서 영상의 첫 번째 프레임은 따로 보관한다. 이후 배경맵을 이용한 수정단계에서 첫 번째 프레임과 이후 프레임들을 이용하여 배경맵을 만들고 만들어진 배경맵을 이용하여 현재 영상을 보정한다. 이후 보정된 영상은 선화단계에서 평균 이동 알고리즘을 이용하여 영상을 색상 기반으로 분할한다. 다음 절에서는 평균 이동 알고리즘과 배경맵 기반 채색 보정에 대해 설명할 것이다.

3.1.1 정지 영상 기반 채색

일반적인 이미지는 여러 가지 색상을 가지고 있으며 그에 따라 외곽선이 검출된다. 본 논문에서는 영역별 색상 단일화를 하기 위해 영상을 색상단위로 단순화하여 분할 할 수 있는 평균 이동 알고리즘을 이용하였다.

평균 이동 알고리즘[12]은 확률밀도함수를 이용한 암벽 등반 탐색 방법의 일종이다. 평균 이동은 어떤 데이터의 피크(peak) 또는 무게중심을 찾는 한 방법으로서, 현재 자신의 주변에서 가장 데이터가 밀집된 방향으로 이동한다. 이러한 작업을 반복하여 시행하면 윈도우 내에 극대 값이 존재하는 경우 그 점으로 수렴하게 된다. 이후 극대 점에서의 컬러 값을 현재 위치의 픽셀 값으로 변경하면 공간 영역에서의 컬러 값을 균일하게 할 수 있다.

3.1.2 비디오 기반 채색 보정

(1) 비디오 기반 채색의 문제점

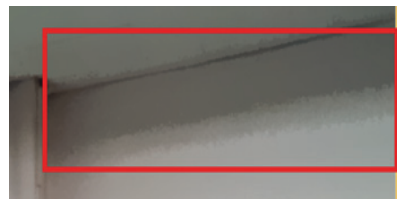


그림 3. 실시간 비디오 영상의 문제점
Figure 3. The problem of the realtime video image

카툰 표현을 하기 위해서는 영역별 색상 단일화 알고리즘을 사용하게 되는데 고정된 이미지가 아닌 실시간 영상에 적용할 경우 문제점이 발생하게 된다. <그림 3>을 보면 영역별 색상 단일화 과정에서 빛에 영향을 받는 색상의 경계 부분이 난잡하게 일렁이게 된다. 블러링 작업을 통해서 어느 정도 개선을 할 수 있지만 일렁이는 것은 마찬가지이다. 이러한 문제점을 해결하기 위해 배경맵을 이용하여 영상을 보정한다.

(2) 배경맵

<그림 4-b>는 현재 영상이며 <그림 4-a>는 참조 영상의 첫 번째 프레임을 복사해둔 영상이다. 이후 <그림 4-a>와 <그림 4-b>를 픽셀 단위로 비교하여 배경맵을 만든다. <그림 4-c>는 손을 위 아래로 몸을 좌우로 조금씩 움직였을 때 만들어진 배경맵이다. 배경맵을 만드는 방법은 원본 영상과 참조 영상을 비교하여 픽셀 값이 다를 경우 참조 영상의 해당 픽셀 값을 원본 영상의 픽셀 값으로 바꾸고 배경맵의 해당 픽셀 값(배경맵의 상태 정보: 0 배경, 1이상 전경)을 증가시켜준다. 만약 원본 영상과 참조 영상의 픽셀 값이 70% 이상 차이가 날 경우 참조 영상으로 해당 시점의 원본영상을 저장하고 배경맵을 초기화 시킨다.

$$D_i(x, y, t) = |I(x, y, t) - R(x, y)|$$

$$M_B(x, y, t) = \begin{cases} M_B(x, y, t-1) + 1 & \text{if } D_i(x, y, t) \geq Th_\beta \\ M_B(x, y, t-1) & \text{if } D_i(x, y, t) < Th_\beta \end{cases} \quad (1)$$

본 연구에서는 배경맵을 만들기 위해 수식 1을 사용하였다. $D_i(x, y, t)$ 는 현재 프레임 $I(x, y, t)$ 와 첫 프레임 $R(x, y)$ 을 뺀 값의 절댓값이다. $M_B(x, y, t)$ 는 배경맵이며 만약 $D_i(x, y, t)$ 가 지정된 임계값 Th_β 보다 클시 전경으로 처리되고 $M_B(x, y, t-1)$ 의 값에서 1만큼 더해준다. 그리고 $D_i(x, y, t)$ 가 지정된 임계값 Th_β 보다 작을시 배경으로 처리되고 $M_B(x, y, t)$ 의 값은 바뀌지 않는다.

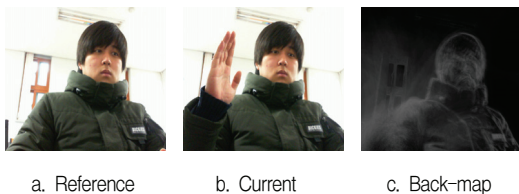
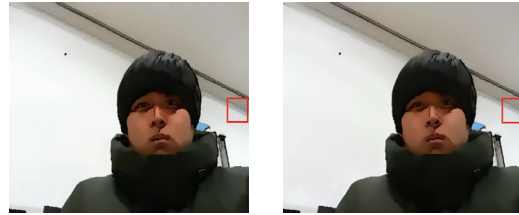


Figure 4. The process of back-map calculation





<그림 5-b>는 배경맵을 적용한 영상이며 <그림 5-a>는 배경맵을 적용하지 않은 영상이다.



a. Still-image painting b. Video painting
 그림 5. 정지 영상 채색과 배경맵 기반 비디오 채색의 비교
 Figure 5. The comparison of still-image painting and back-map based video painting

아래 <표 1>은 이전영상과 현재 영상, 현재 영상과 다음 영상의 픽셀차를 배경맵을 사용했을 때와 배경맵을 사용하지 않았을 때를 비교한 표이다. 각 그림은 빛에 영향을 받는 영역을 잘라낸 영상이다. 배경맵을 사용하지 않은 이전 영상과 현재 영상의 픽셀차는 23877로 나왔으며 현재 영상과 다음 영상의 픽셀차는 29392로 나왔다. 하지만 배경맵을 사용한 이전 영상과 현재 영상의 픽셀차는 5652, 현재 영상과 다음 영상의 픽셀차는 3029로 배경맵을 사용했을 때와 사용하지 않았을 때의 차이가 크을 알 수 있다.

표 1. 픽셀 차이 비교
 Table 1. The comparison of pixel difference

	pixel difference (without back-map)	pixel difference (with back-map)
Current - Reference	 23877	 5652
Reference - Next	 29392	 3029

3.2 선화

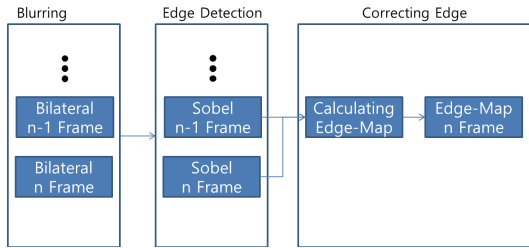


그림 6. 선화 단계의 시스템 흐름도
Figure 6. System flowchart of inking step

<그림 6>은 선화 단계의 상세한 시스템 흐름도이다. 채색 단계의 결과 영상을 블러링 단계에서 양방향 알고리즘을 이용하여 블러링을 해준다. 이후 외곽선 검출 단계에서 소벨 외곽선 검출을 이용하여 외곽선을 추출한다. 소벨 외곽선 검출을 통해 추출된 영상은 백업하여 엣지 수정 단계에서 이전 영상의 소벨 프레임과 현재 영상인 소벨 프레임 두 영상의 차영상 즉 엣지맵을 만든다. 다음 절에서는 정지영상 기반 선화와 비디오 기반 선화 보정에 대해 설명할 것이다.

3.2.1 정지 영상 기반 선화

채색 단계에서 평균 이동 알고리즘을 거친 후 먼저 영상에 블러링을 하여 영상을 보정해준다. 블러링 단계에서는 양방향 필터[13]과 가우시안[14]을 이용하였다. 카툰 렌더링 표현을 위해서는 외곽선 검출이 필수적인 요소이다. 본 논문에서는 외곽선 검출 알고리즘[15] 중 캐니, 샤프닝, 소벨 외곽선 검출 알고리즘을 비교하였고 그중 카툰렌더링의 외곽선에 적합한 소벨을 선택하였다.

(1) 블러링

<그림 7-a>는 평균 이동 알고리즘을 적용 이후

양방향 필터를 적용한 이미지이며 <그림 7-b>는 평균 이동 알고리즘을 적용 이후 가우시안을 적용한 이미지이다. 그리고 <그림 7-c>와 <그림 7-d>는 각각 <그림 7-a>와 <그림 7-b>에 소벨 외곽선 검출을 한 결과이다. 결과적으로 <그림 7-d>는 <그림 7-c>보다 외곽선의 두께가 두껍지만 이미지가 선명하지 못하고 흐려져 사용하기에 부적합 하였다. 때문에 블러링 단계에서는 양방향 필터를 사용하였다.

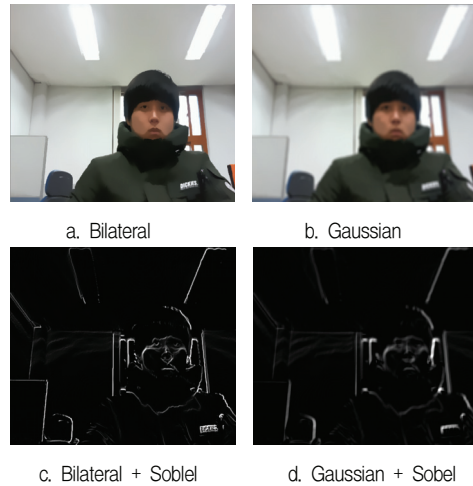


그림 7. 가우시안과 양방향 필터의 비교
Figure 7. The comparison of gaussian and bilateral filter

(2) 외곽선 검출

<그림 8-a>는 캐니 외곽선 검출, <그림 8-b>는 샤프닝, <그림 8-c>는 소벨 외곽선 검출의 결과이다. 캐니 외곽선 검출의 경우 외곽선 검출은 잘 되지만 두께가 너무 얇았으며 샤프닝의 경우는 두께도 얇고 외곽선이 선명하게 나오지도 않았다. 마지막으로 소벨 외곽선 검출은 외곽선 검출이 잘 되며 캐니와 샤프닝에 비해 두께가 두꺼워 카툰 렌더링의 외곽선에 적합하였다. <그림 8-d-f>는 평균 이동 알고리즘을 적용한 영상에 외곽선을 입힌 영상이다. 소벨 외곽선 검출을 입힌 <그림 8-f>가

다른 영상들 보다 외곽선이 선명한 것을 확인할 수 있다.

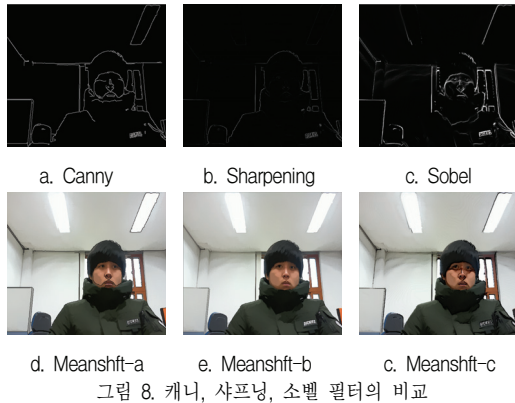


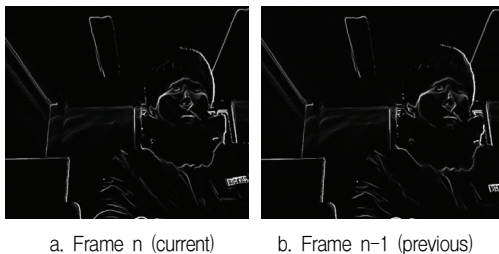
그림 8. 캐니, 샤프닝, 소벨 필터의 비교
Figure 8. The comparison of canny, sharpening and sobel filter

3.2.2 비디오 기반 선화 보정

(1) 비디오 기반 선화의 문제점

정지영상에서 선화 작업을 하면 움직이는 물체가 없어 외곽선 검출의 결과에 영향을 주지 않지만, 비디오 기반의 영상에서 선화 작업을 할시 카메라 이동과 물체의 이동에 외곽선 검출이 영향을 받게 된다. 물체가 천천히 움직인다면 영향이 많지 않지만 빠르게 움직인다면 외곽선 검출의 결과가 정확하지 않아 이러한 문제점을 보완하기 위해 엣지맵을 이용하였다.

(2) 엣지맵



c. Edge-map d. Frame n + edge-map
그림 9. 엣지맵 생성 과정

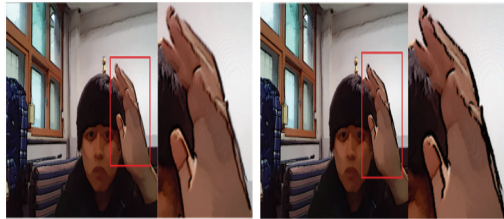
Figure 9. The Process of edge-map generation

엣지맵은 현재 프레임의 영상과 이전 프레임의 영상을 뺀 영상을 말한다. 엣지맵을 만드는 수식은 수식 2와 같다. M_E 는 엣지맵, E 는 외곽선 검출 결과 영상을 뜻한다. 엣지맵 $M_E(x, y, t)$ 는 현재 프레임의 외곽선 검출 픽셀 값과 이전 프레임의 외곽선 검출 픽셀 값을 뺀 값의 절댓값을 뜻한다.

$$M_E(x, y, t) = |E(x, y, t - 1) - E(x, y, t)| \quad (2)$$

<그림 9-c>는 현재 프레임 영상인 <그림 9-a>와 이전 프레임 영상인 <그림 9-b>를 빼준 영상에 임계치를 준 영상이다.

<그림 9-c>의 임계치 값은 50이며 50보다 낮은 픽셀은 0으로 이외의 픽셀은 255로 바뀌어 있었다. <그림 9-d>는 현재 프레임 영상인 <그림 9-a>와 임계치 값 50을 적용한 엣지맵인 <그림 9-c>를 합한 영상이다. <그림 10-a>와 <그림 10-b>는 손을 좌우로 움직이는 영상을 캡처한 것이며 <그림 10-a>는 엣지맵 보정을 적용하지 않은 영상이고 <그림 10-b>는 엣지맵 보정을 적용한 영상이다. <그림 10-a>와 <그림 10-b>를 비교해보면 <그림 10-a>가 <그림 10-b>에 비해 움직이는 물체의 외곽선이 더욱 선명하게 나타나는 것을 확인할 수 있다.



a. Still-image inking b. Video Inking
 그림 10. 엣지맵 생성 과정

Figure 10. The Process of edge-map generation

4. 연구결과



a. Back-map b. Edge-map



c. without back and edge-map d. with back and edge map
 그림 11. 제안된 비디오 기반 카툰 렌더링의 결과

Figure 11. The result of suggested video based cartoon rendering.

<그림 11-a>는 배경맵 영상이며 <그림 11-b>는 엣지맵 영상이다. <그림 11-c>는 배경맵으로 원본 영상 보정과 엣지맵으로 선화 보정을 해주지 않은 영상이다. <그림 11-d>는 배경맵을 이용한 원본 영상 보정 후 평균 이동 알고리즘으로 채색, 블러링과 소벨 외곽선 검출로 외곽선 검출 그리고 엣지맵으로 선화 보정을 해준 결과 영상이다. <그림 11-c>와 <그림 11-d>를 비교 하였을 때 움직이는 영상에서 배경맵을 사용 시 물체의 외곽선이 더욱 선명하게 나타나는 것을 확인할 수 있다.

5. 결론 및 향후 연구

본 논문에서는 일반적인 정지영상 기반 카툰 렌더링이 아닌 비디오 기반 실시간 카툰 렌더링을 구현 하였으며 비디오 기반 카툰 렌더링의 일링이는 문제점을 배경맵을 이용해 해결하는 방법을 제안 하였다. 또한 움직이는 물체의 선화 작업을 엣지맵을 이용해 보정하는 방법을 사용 하였다.

향후 연구로는 제안된 카툰렌더링에 속도 향상 기법을 적용한 실시간 렌더링 성능 평가가 필요하다. 또한, 적용 평균 이동 알고리즘이 사람의 얼굴에 적용 될 경우 결과가 좋지 않게 나오는데 평균 이동 알고리즘을 보완하여 사람의 얼굴 또한 자연스러운 카툰 표현이 되도록 연구할 예정이다.

References

- [1] B. Gooch, and A. Gooch, *Non photorealistic rendering*, A K Peters, Ltd. 2001.
- [2] P. Decaudin, *Cartoon-looking rendering of 3D-scenes*, Research Report INRIA, 1996.
- [3] S. Dietrich, *Cartoon rendering on the geforce 256*, NVidia Corporation, 1999.
- [4] J. Lander, *Shades of disney: Opaqing a 3D world*, Game Developer, Mar. 2000.
- [5] C. S. Marshall, *Cartoon rendering: Real-time silhouette edge detection and rendering*, Game Programming Gems 2, Charles River Media, pp. 436-443, 2001.
- [6] S. Chenney, M. Pingel, and R. Iverson, *Simulating cartoon style animation*, Non-Photorealistic Animation and Rendering 2002 (NPAR '02), Annecy, France, Jun. 2002.
- [7] K. Anjyo, and K. Hiramitsu, *Stylized highlights for cartoon rendering and animation*, IEEE Computer Graphics and Applications, Vol. 23,

No. 4, pp. 54-61, 2003.

- [8] L. Petrovic, B. Fujito, L. Williams, and A. Finkelstein. *Shadows for cel animation*, Proceedings of SIGGRAPH 2000, Computer Graphics Proceedings, pp. 511-516, 2000.
- [9] M. A. Kowalski, L. Markosian, J. D. Northrup, L. Bourdev, R. Barzel, L. S. Holden, and J. F. Hughes, *Art-based rendering of fur, grass, and trees*, Proceedings of the 26th annual conference on Computer graphics and interactive techniques, pp. 433-438, Jul. 1999.
- [10] A. Lake, *Cartoon rendering using texture mapping and programmable vertex shaders*, Game Programming Gems 2, Charles River Media, pp. 444-451, 2001.
- [11] J. Claes, F. Di Fiore, Vansichem, G. Vansichem, and F. Van Reeth, *Fast 3D cartoon rendering with improved quality by exploiting graphics hardware*, Proceeding of Image and Vision Computing, pp. 13-18, 2001.
- [12] Comaniciu, Dorin, and P. Meer, *Mean shift: A robust approach toward feature space analysis*, IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 24, No. 5, pp. 603-619, 2002.
- [13] C. Tomasi, and R. Manduchi, *Bilateral filtering for gray and color images*, IEEE Transactions on Computer Vision, pp. 839-846, 1998.
- [14] R. A. Haddad, and A. N. Akansu, *A class of fast gaussian binomial filters for speech and image processing*, IEEE Transactions on Acoustics, Speech and Signal Processing, Vol. 39, pp. 723-727, Mar. 1991.
- [15] D. Ziou, and S. Tabbone, *Edge detection techniques: An overview*, International Journal of Pattern Recognition and Image Analysis,

Vol. 8, No. 4, pp. 537-559, 1998.

배경맵과 옛지맵을 이용한 비디오 기반 카툰 렌더링

장재건, 박현진, 류승택

한신대학교 컴퓨터공학부

요 약

카툰 스타일 렌더링의 목적은 코믹 북과 카툰 무비와 같이 전통적인 만화 스타일의 형태를 표현하는 것이다. 카툰 렌더링의 핵심은 검은 외곽선 표현(선화)과 내부 영역의 단일 색상 표현(채색)이다. 본 논문에서는 프레임 차이를 이용한 영상 기반 카툰 렌더링을 제안한다. 비디오 기반 카툰 렌더링은 시간 일관성에 의해 발생하는 문제가 있다. 이러한 문제들 중 하나는 움직이는 물체의 정확한 외곽선을 그리는 것이 어렵다는 점이다. 그리고 그림의 경계 지역에 물결치는 효과가 조명 변화에 의해 나타난다. 물결 효과를 해결하기 위해 채색단계에서 현재 프레임과 이전 프레임 사이의 차이를 이용하여 배경맵을 적용하였다. 보정된 프레임은 배경맵을 이용하여 참조 프레임과 현재 프레임을 브렌딩하여 만들 수 있다. 이렇게 보정된 프레임은 평균 이동 알고리즘을 이용하여 색상 기반 영역 분할된 프레임으로 변환된다. 또한, 부정확한 외곽선을 개선하기 위해, 선화 단계에서 현재 옛지맵에 현재와 이전 옛지의 차영상을 첨가한 옛지맵을 적용하였다. 블러된 프레임은 색상 분할된 프레임에 양방향 알고리즘을 적용하여 생성된다. 정확한 외곽선 옛지를 추출하기 위해 소벨 외곽선 검출 알고리즘을 적용하였다. 최종 결과 영상은 보정된 색상 분할 프레임부터 옛지맵을 차감하여 생성될 수 있다.

감사의 글

본 논문은 중소기업청에서 지원하는 2015년도 산학연협력 기술개발사업(기업부설연구소 신규설치)(No. C0261447)의 연구수행으로 인한 결과물임을 밝힙니다.



Jae Khun Chang received his B.S. degree from Hanyang University in 1985. He had received his M.S. degree from Department of Computer and Information

Science in New Jersey Institute of Technology, and Ph.D. from Department of Computer Science in University of South Carolina in 1989 and 1997 respectively. He is currently in School of Computer Engineering, Hanshin University, Korea. His interests are computer vision, ITS, image processing, pattern recognition, and motion tracking.

E-mail address: jchang@hs.ac.kr

rendering, non-photorealistic rendering.

E-mail address: stryoo@hs.ac.kr



Hyeon Jin Park received his B.S. degree from Hanshin University in 2015. He is currently a graduate student in Department of Computer Engineering, Hanshin

University, Korea. His interests are non-photorealistic rendering and Real-time rendering

E-mail address: phj3372@hs.ac.kr



Seung Taek Ryoo received the B.S. and M.S. degrees in Computer Science Engineering from ChungAng University in 1996 and 1998, respectively. He had received his Ph.D.

in Department of Image Engineering, Graduate School of Advanced Imaging Science, Multimedia and Film from ChungAng University in 2002. He is currently an associate professor in School of Computer Engineering, Hanshin University, Korea. His interests are computer graphics, image-based rendering, realtime