

2D 그림에서 3D 객체 변환에 의한 AR 시스템 개발

장 우규¹, 신 봉기²
부경대학교 컴퓨터공학부

¹wookyu93@gmail.com, ²bkshin@pknu.ac.kr

Development of an AR System through 3D Object Conversion from 2D Drawings

Woo-Kyu Jang, Bong-Kee Shin
Div. of Computer Engineering, Pu-Kyong National University

요 약

2D 객체 그림으로부터 3D 객체로 변환하는 것은 저차원 신호로부터 고차원 공간 정보를 추론하는 문제이다. 본 연구에서는 특정 주제 내에서 미리 구축해 놓은 3D 객체 모델 기반으로 해결하는 ill-posed 문제이다. 사용자의 2D 그림과 가장 유사한 베이스 모델을 판별한 후, 해당 모델을 기반으로 3D 객체를 구성하고 채색 과정을 거쳐 완성한다. 본 연구에서는 이를 기반으로 변환한 객체를 증강 현실 환경에서 구현한다.

1. 서론

컴퓨터 비전(Computer Vision, CV)은 기계가 영상으로부터 정보를 추출하고, 분석하여 특정 작업을 수행하게 하는 기술 분야이다. 현재, CV 기술은 다양한 산업분야에서 활용되어 활발한 연구가 진행 중이지만, 여전히 많은 도전과제를 가지고 있다.

그 가운데, 2D 그림에서 3D로 변환하는 문제는 상당히 흥미로운 주제 중 하나이며, 이는 본 연구의 목표이다.

본 연구는 2D 그림으로부터 3D 객체 모델로 구성하고, 이를 증강현실(Augmented Reality, AR)[1] 환경에서 구현하는 것이다.

2. 시스템 구성

본 논문에서는 다음과 같은 절차를 가지는 시스템을 제안한다.

- 1) 3D 베이스 모델 선정 및 객체 정보 구축.
- 2) 2D 그림 영상 처리 및 특징 추출.
- 3) 최적 베이스 모델 선정.
- 4) 선정된 베이스 모델에 텍스처 적용.
- 5) 결과 모델을 AR 환경에 적용.

이 과정은 Figure 1 으로 도식화되어 있다.

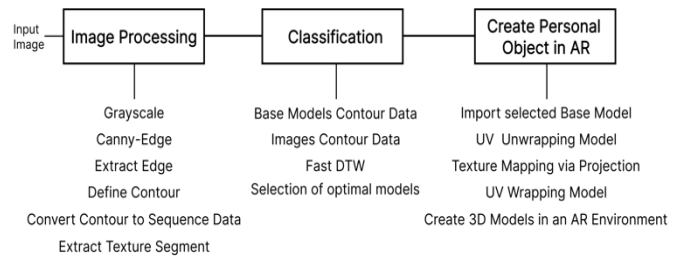


Figure 1

2.1 영상처리(Image Processing): 특징 추출

스케치한 그림의 2D 영상을 입력으로 한다. 이를 명도 영상으로 변환한다. Canny-Edge[2] 검출 방법을 이용해 그림 속 핵심 객체의 선을 추출하고 이를 테두리로 정의한다. 정의한 테두리의 크기를 재고하고, 이를 고려하여 적절한 개수로 방향 선분 계열로 근사 교정한다. 가장 왼쪽 픽셀을 시작점으로 하여 시계방향으로 나누어진 각 선분은 'Figure 2' 그림과 같이 하나의 계열로 가정한다.

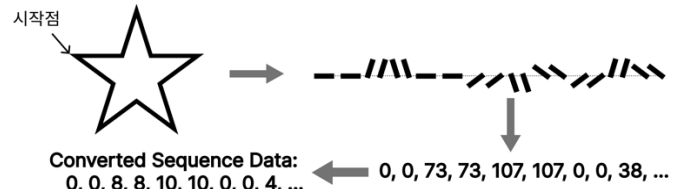


Figure 2

이후 각 선분은 직선 위에서의 각도 값으로 변환하고, 계산 복잡도를 줄이기 위해 각도 값을 '2' 단위로 치환하여 분류를 위한 데이터로 저장한다.

한편 앞서 추출한 외곽선을 기준으로 그림 내부의 세부 영역을 추출하여 텍스처로 정의한다.

2.2 분류

2D 그림으로부터 변환된 시계열 데이터와 3D 객체의 측면도로부터 변환된 시계열 데이터들을 비교하기 위해 Fast-DTW 를 사용한다.

DTW(Dynamic Time Warping)는 다양한 길이와 시간 왜곡을 갖는 시계열 데이터간 유사성을 판단하기에 적합하지만, 계산 복잡도가 높은 DTW 의 약점을 보완하기 위해 Fast-DTW[3]를 사용한다.

2.3 3D 모델 개인화 및 오브젝트 생성

텍스처를 3D 모델에 매핑하기 위해 UV Wrapping[4] 기술을 활용한다. UV Wrapping 은 3D 모델의 표면을 polygon 에 따라 전개하고, 그 전개도 위에 텍스처를 projection 하여 매핑하는 기법이다. 이 과정은 3D 모델링 툴인 Blender 의 API 스크립트를 작성해 자동화한다.

2.4 서버-클라이언트와 증강 현실

증강 현실 구현을 위해 필요한 자원을 쉽게 확보할 수 있고, 사용자 접근성과 편의성을 위해 Android Application(App)을 사용하여 클라이언트를 구성한다.

이로써 하나의 응용 소프트웨어에서 그림 입력과 증강 현실을 모두 구현하게 된다.

또한, 사용자 기기의 자원 절약과 빠른 계산을 위해 서버-클라이언트 방식으로 구현한다.

서버는 JAVA Spring Boot[5]를 이용해 구현한다. 이로 인해 사용자 디바이스의 자원사용량을 절약하고, 빠른 키프레임으로 실시간 응답이 가능한 장점이 있다.

3. 결과 및 토의

결과적으로 그림을 분류하여 가장 유사한 베이스 모델을 선정하는 과정은 성공적으로 수행되었다. 다만, 그림 속 객체의 텍스처를 왜곡 없이 정확하게 3D 모델에 매핑하는 작업은 아직 미완 상태이다.

Figure 3 은 입력된 그림의 예를 보여준다. 현재는 그림을 분류하는 프로세스와, 객체의 텍스처를 추출하는 프로세스를 Figure 3 와 같이 분리하여 동작하는 수준으로 구현되었다.

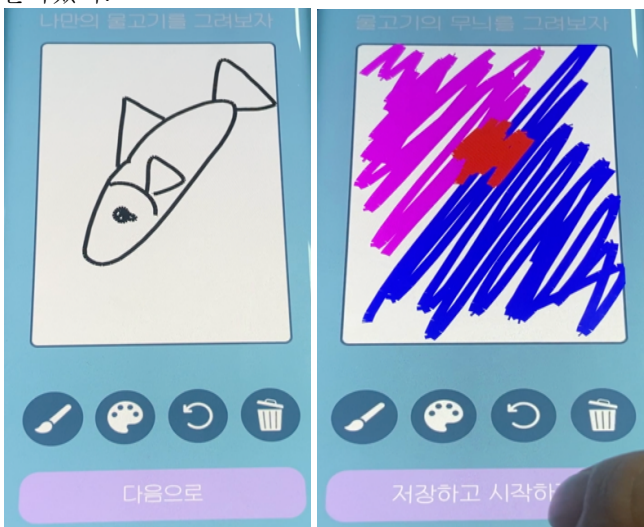


Figure 3



Figure 4: 결과

프로세스를 분리하여 수행한 결과, Figure 4 와같이 유사한 모델을 선정하고, 텍스처를 매핑하는 프로세스는 성공적으로 수행되었다.

4. 결론

본 연구에서는 3D 베이스 모델을 활용하여 2D 그림과 가장 유사한 모델을 찾아내고, 이를 기반으로 3D 객체로 변환하는 알고리즘을 제안한다.

그렇지만, 본 연구는 2D 그림에서 3D 객체로의 변환 문제에 대한 새로운 방향성을 제시하였으며, 추후 연구에서는 텍스처 매핑의 정확성을 향상시키는 방법에 대한 탐구가 필요하다고 판단한다.

참고문헌

- [1] Ronald T. Azuma; A Survey of Augmented Reality. Presence: Teleoperators and Virtual Environments 1997.
- [2] J. Canny, "A Computational Approach to Edge Detection," in IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. PAMI-8, no. 6, pp. 679-698, Nov. 1986.
- [3] Salvador, S., & Chan, P. (2004). FastDTW: Toward Accurate Dynamic Time Warping in Linear Time and Space. Dept. of Computer Sciences, Florida Institute of Technology, Melbourne, FL 32901. Pages 1-4.
- [4] Igarashi, T., & Cosgrove, D. (2001). Adaptive Unwrapping for Interactive Texture Painting. In Proceedings of the 2001 symposium on Interactive 3D graphics (I3D '01), 209-216.
- [5] Walls, C. (2015). Spring Boot in Action. [Manning Publications Co.]: Manning Publications.