만화 영상에서 말풍선의 자동 생성 방법*

류동성[O] 천봉경 박규태 조환규
부산대학교 컴퓨터공학과
{dsryu, bkchun, ktpark}@pearl.cs.pusan.ac.kr

Automated Generation of Word Balloons in Comics

DongSung Ryu[O] BongKyung Chun KyuTae Park and HwanGue Cho
Department of Computer Engineering, Pusan National University

요 약

일반적으로 말풍선은 만화에서 대사와 주인공을 연결하는 매개체 역할을 한다. 또한 말풍선은 대사가 진행되는 순서를 고려하여 배치되기 때문에, 말풍선의 위치는 줄거리의 전개 과정을 표현하는 역할을 하기도 한다. 그러므로 말풍선을 생성하고 배치하는 작업은 중요한 작업이며, 이러한 작업은 주로 만화가의 수작업을 통해서 이루어진다. 본 논문에서는 만화가의 수작업으로 진행되는 말풍선의 생성과 배치 작업을 자동으로 수행하기 위한 방법에 관해 논의한다. 이를 위해서, 6가지 종류의 말풍선을 모델링하였으며, 이러한 말풍선들을 배치를 위해서, EPFLP 기반의 휴리스틱 알고리즘을 제시하였다. 또한, 말풍선의 참조점과 말꼬리의 방향을 고려해서, 수작업으로 생성되던 말꼬리를 자동으로 생성하는 방법을 제안한다.

Abstract

Generally, word balloon have played a role to connect the script with character in comics. The location of word balloons depicts the process of story in comics, because they are located by reading order. Therefore, it is very important works to generate and place word balloons, these work usually is processed manually by comic writer's. In this paper, we discuss the automated generation and placement of word balloon. For this, we modeled 6 kinds of word balloons. And these word balloons are placed by heuristic method based on EPFLP. We also generate the tail of word balloon automatically by considering the direction and reference points of word balloon.

키워드(국문) : 말풍선, 말풍선의 꼬리, 말풍선 배치
Keywords(English) : Word balloon, Tail of word balloon, Word balloon placement

1. 서론

만화에서 말풍선은 대사와 주인공 사이의 매개체 역할을 하며, 대사가 읽혀지는 순서가 말풍선의 위치에 따라 결정된다. 그러므로 만화에서 말풍선을 생성하고 배치하는 작업은 아주 중요하며, 이러한 작업은 대부분 만화가들의 관점에 따라 수작업으로 진행된다. 본 논문에서는 만화가의 수작업으로 이루어지는 말풍선의 생성과 배치 작업을 자동으로 처리하기 위한 방법에 관해서 주로 논의한다.

만화는 비사실적인 렌더링의 한 분야로써, 오랫동안 연구되고 있다. 최근까지 만화와 연관된 대부분의 연구는

* 이 논문은 정보통신부 "비사실적 에니메이션 기술 개발" 과제 지원에 따른 결과입니다.

카투닝과 같은 이미지 영상의 추상화 기법에 대부분 초점을 맞추어 진행되어 왔으며, 말풍선과 관련된 연구는 주로 말풍선을 배치하기 위한 방법에 관해 진행되고 있다. David[1]는 1999년에 말풍선을 이용한 온라인 채팅 시스템인 Comic chat을 발표하였다. Comic chat은 먼저 사용자의 캐릭터를 생성한 후, 캐릭터 사이의 대화를 말풍선으로 표현하는 시스템이다. 이 시스템은 Routing Channel 이라는 개념을 사용하여, 말풍선들의 꼬리가 서로 교차되지 않도록 배치하였다. Chun, B. K. et al.[2]은 PFLP (Point Feature Label Placement) 기반의 레이블 배치 기법을 응용하여, 말풍선의 읽는 순서(Reading Order)에 따라 각각의 말풍선들이 배치될 수 있는 영역을 제한한 후, 휴리스틱 기법을 사용하여 말풍선의 배치 문제를 해결하였다.

2. 본론

2.1. 다양한 종류의 말풍선

말풍선은 대사의 분위기나 현재 상황에 따라 여러 가지 형태로 표현된다. 그림 1은 본 논문에서 생성한 6가지 형태의 말풍선이다. 이러한 스타일의 말풍선을 렌더링하기 위해서, 타원의 수식과 내접원 및 외접원으로 구성된 모델(그림 2)을 사용하였다. 타원 형태의 모델에 사용된 수식은 수식 1과 같으며, d 값을 이용해서 타원의 모서리 부분의 둥근 정도를 조절한다(그림 1의 (a)와 (d) 말풍선).

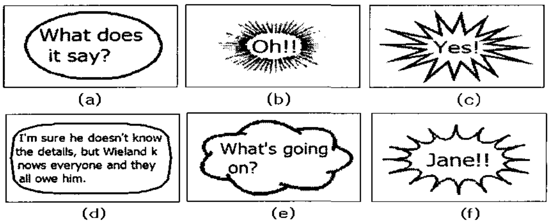

그림 1 다양한 종류의 말풍선들. (a) 기본 말풍선(d=2) (b) 느낌 말풍선. (c) 강조 말풍선. (d) 모서리가 둥근 사각형 말풍선(d=4). (e) 생각 말풍선. (f) 외침 말풍선.

$$\left(\frac{x}{a}\right)^d + \left(\frac{x}{b}\right)^d = 1 + \alpha \qquad (1)$$

그림 2는 그림 1의 (b), (c), (e) 그리고 (f) 말풍선들을 모델링하기 위해, 사용된 모델이다. 본 논문에서는 그림 1의 (b) (c) (e) (f) 와 같이 내접원과 외접원으로 구성하여 모델링 되는 말풍선을 외침 계열의 말풍선이라 명명하였다. 외침 계열의 말풍선들은 내접원의 정점들(IP_i)과 외접원의 정점들(CP_i)을 말풍선의 세부적인 스타일을 제어하기 위한 제어점으로 사용한다. 만약 각 제어점들의 사이각($\overrightarrow{D_i}$)을 작게 설정하면, 말풍선의 주위가 조밀하게 되며, 내접원의 지름($\overrightarrow{G_i}$)을 크게 하면, 대사를 표현하는 공간이 넓은 말풍선이 형성된다.

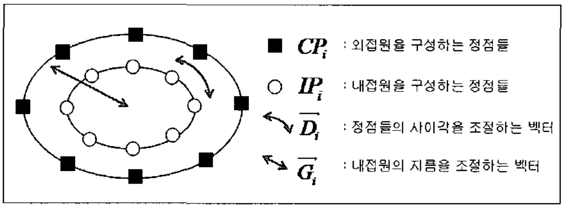

그림 2 외침 계열의 말풍선을 렌더링하기 위해 사용한 모델.

2.2. 말풍선의 자동 배치 방법

말풍선의 배치 작업은 만화가의 주관에 따라 대부분 수작업으로 수행한다. 그러나 대부분의 독자들은 만화의 한 장면에서 오른쪽 보다는 왼쪽에, 그리고 아래쪽 보다는 위쪽에 위치한 말풍선의 대사를 먼저 읽게 된다[2]. 즉 말풍선들이 읽는 순서인 Reading Order에 따라 배치되는 것이다.

Chun, B. K. et al.은 확장된 특징점 기반의 레이블 배치(EPFLP: Extended Point Feature Label Replacement) 방법과 Qlayout 평가 함수를 사용한 휴리스틱 기법을 사용하여, 말풍선을 읽는 순서(Reading Order)에 맞게 자동으로 배치하는 방법을 제안하였다. 본 논문에서는 Chun, B. K. et al.[1]이 제안한 휴리스틱 방법에 가상의 힘을 추가하여, 말풍선들을 좀 더 확장된 방향으로 배치 가능하게 하였다. 수식 2는 본 논문에서 추가한 가상의 힘이며, 두 점전하 사이에 작용하는 힘인 쿨롱의 법칙과 유사하게 모델링하였다. 쿨롱의 법칙에서 두 점전하의 전하량 대신에, $\overrightarrow{S_i}$ 벡터를, 점전하의 위치 대신에 말풍선의 위치(W_i)를 사용하였다. 여기서 $\overrightarrow{S_i}$ 벡터는 만화에서 i-th 말풍선의 넓이를 크기로 하고, 말풍선의 중심에서 참조점들의 무게중심 방향의 벡터입니다. 즉, 넓이가 큰 말풍선일 경우 큰 힘이 작용하며, 참조점들의 밀집도가 높은 곳에 있을수록, 큰 힘이 작용한다. 여기서 $\vec{f}$는 i-th 말풍선에 작용하는 가상의 힘에

대한 방향을 표현하며, 수식 3과 같이 설정하였다. 수식 3의 결과 값은 정규화된 벡터이며, i-th 말풍선의 위치가 다른 말풍선에 미치는 영향을 고려하여, 가상의 힘이 작용하는 방향을 계산하였다.

$$force\ \vec{F}(WB_i) = \alpha \frac{\sum_{j=0}^{N} \vec{S_i} \cdot \vec{S_j}}{\sum_{j=0}^{N} (W_i - W_j)^2} \vec{f_i} \qquad (2)$$

$$\vec{f_i} = normalize\left(\sum_{j=0}^{N} \overrightarrow{W_i W_j}\right) \qquad (3)$$

알고리즘 1은 본 논문에서 제안한 가상의 힘을 Chun, B. K. et al.의 방법에 적용하여, 수정한 알고리즘이다. 이 알고리즘에서 w.b., r.p., v.b.와 i.b.는 각각 말풍선(word balloon), 참조점(reference point), 말풍선의 가상경계영역(virtual boundary) 그리고 만화의 경계영역(image boundary)을 의미한다. 여기서 각 말풍선이 참조하는 참조점(r.p)은 사용자가 수작업으로써 적당한 위치를 지정하게 하였다.

```
repeat
  select the farthest w.b. from its own r.p.
  move w.b. toward its own r.p.
  calculate force of each w.b.
  apply force to each w.b.
  make v.b. of each w.b.
  if v.b meet i.b. then
    calculate the v.b. area out of i.b.
    convert w.b. into boundary w.b.
  end-if
  calculate QLayout
until RO regulation is O.K. and QLayout is more bigger
than old QLayout.
```

알고리즘 1. 말풍선을 자동으로 배치하기 위한 알고리즘.

2.3. 말풍선 꼬리의 자동 생성 방법

말풍선의 꼬리는 말풍선이 어떤 캐릭터에 의해서 말해졌는지를 표현하는 매개체 역할을 한다. 만약 만화의 캐릭터들이 서로 모여 있을 경우, 단순한 방법으로 말꼬리를 생성한다면, 각각의 말풍선이 어떤 캐릭터를 묘사하는지 모호한 경우가 발생한다. 그림 3 (a)의 경우, WB_i 말풍선의 참조점은 말꼬리의 방향이 모호하게 표현되기 때문에, R_i인지 R_j인지 이해하기가 힘들다. 즉 WB_i 말풍선의 대사 내용이 주인공 A의 것인지 주인공 B의 것인지 모호한 경우를 표현하고 있다. 이러한 경우 그림 (b)와 같이 대부분의 만화에서는 말꼬리의 형태와 위치를 이동하거나 곡률을 설정함으로써, 참조점의 모호함을 해결한다. 그러므로 본 논문에서는 말꼬리가 참조점을 좀 더 명확하게 표현하기 위해서, 곡률을 설정하고, 이에 따라 말꼬리의 위치 및 형태를 변경하여, 말풍선의 꼬리를 자동으로 생성한다.

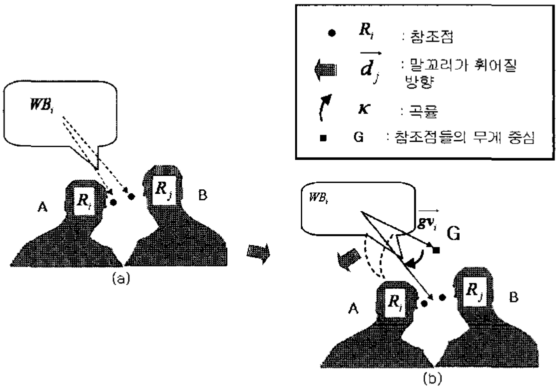

그림 3 말풍선에서 말꼬리를 생성할 때, 고려해야 할 요소들. WB_i, G, $\vec{gv}$는 각각 i-th 말풍선, 참조점들의 무게 중심, 그리고 말풍선의 중심에서 G까지의 벡터를 각각 의미한다.

본 논문에서는 곡률을 참조점들 사이의 밀집도와 말꼬리의 방향을 고려하여 계산한다. 말풍선 꼬리의 곡률은 말꼬리가 휘어지는 양을 의미하며, 수식 (4)~ (6) 에 의해 계산된다.

$$\delta\theta = \kappa_i \theta_{ext} \qquad (4)$$

수식 (4)에서 $\delta\theta$는 곡률(κ_i)에 의해서 실제로 휘어지는 각을 의미하며, θ_{ext} 는 말풍선의 꼬리가 최대로 휘어질 수 있는 최대각을 나타낸다. 본 논문에서 θ_{ext}는 0.5π로 설정한다. 여기서 실제로 말꼬리가 휘어지는 곡률은 수식 (5)와 같이 계산되며, 그 방향은 그림 3과 같이 벡터 $\vec{gv_i}$(WB_i에서 G로 향하는 방향)를 기준으로 참조점(R_i)의 위치가 왼쪽인지 오른쪽인지에 따라 결정된다.

$$\kappa_i = \left(1 - e^{-\alpha(c_i - 0.5)^2}\right) + \beta\theta^2 \frac{|\vec{v_i}|}{\sum_{i=1}^{N} |\vec{v_i}|} \qquad (5)$$

수식 (5)는 곡률의 크기(κ_i)를 결정하는 수식인데, 여

기서 c_i는 수식 (6)과 같이, 참조점 R_i가 다른 참조점에 비해 얼마나 떨어져 있는가를 평가하는 스칼라값이다. 즉, 본 논문에서 제안하는 말꼬리의 자동 배치 알고리즘은 참조점들이 서로 가까이 모여 있을 경우, 참조점을 보다 명확하게 표현하기 위해서, 말꼬리가 휘어지는 정도인 곡률을 더 크게 적용한다. 또한, 수식 (5)에서 θ는 $\vec{v_i}$와 $\overrightarrow{gv_i}$의 사이각을 의미하며, 이것은 말꼬리의 방향을 표현하는 벡터 $\vec{v_i}$가 말풍선의 참조점들의 무게중심(G) 방향인 벡터 $\overrightarrow{gv_i}$와 일치하는 정도를 내적으로 반영한 값이다.

$$c_i = \left(\frac{\sum_{i=1}^{N} |R_i - R_j|}{\sum_{i=1}^{N} \sum_{j=1}^{N} |R_i - R_j|} \right) \qquad (6)$$

그림 4는 본 논문에서 제안한 방법에 따라 말풍선과 말꼬리를 생성하고, 이를 자동으로 배치한 만화의 한 장면이다. 말풍선의 자동 배치나 말꼬리의 자동 배치 기능을 활용하여, 수작업으로 진행되던 작업을 최소화하였으며, 다양한 형태의 말풍선을 보다 쉽게 생성하고 배치할 수 있었다.

3. 결론

본 논문에서는 만화가의 수작업으로 제작되던 말풍선의 생성과 배치 작업을 참조점의 설정이나 대사의 입력과 같이 말풍선을 배치하기 위한 기본적인 정보의 입력만으로써, 말풍선을 보다 편리하게 생성하는 방법을 제안하였다. 이를 위해서, 6가지 스타일의 말풍선을 모델링하였으며, 말풍선의 자동 배치 알고리즘인 휴리스틱 기법[2]에 가상의 힘을 추가하여, 말풍선의 배치 영역을 좀 더 확장하였다. 또한 참조점들과 말풍선의 방향을 고려해서, 말꼬리를 자동으로 생성하였다.

그러나 만화의 경우, 만화가의 주관이나 만화의 분위기에 따라 더 많은 형태의 말풍선 스타일이 존재하며, 사용되는 용도 또한 다양하다. 예를 들어, 만화에서 사용되는 폰트의 경우, 다양한 분위기를 표현할 수 있으며, 여러 가지 만화적인 효과를 적용하여 표현된다. 그러므로 본 논문에서는 향후 연구 과제를 다음과 같이 제시한다.

(a) 의성어와 같이 과장된 스크립트를 표현하기 위한 폰트들을 개발하고, 이를 제어하기 위한 인터페이스를 개발한다.

(b) 만화 주인공에 해당하는 객체를 스스로 찾고, 말풍선 배치시, 객체 영역과 중첩되지 않게 배치하는 알고리즘을 개발한다.

(a)

(b)

그림 4 본 논문에서 제안한 방법에 의해 만들어진 만화 - (a) 영화 24시의 한 장면에 말풍선을 추가한 결과 (b) 영화 친구의 한 장면을 이진화해서 말풍선을 배치한 결과.

참고문헌

[1] David Kurlander and Tim Skelly and David Salesin. *Comic chat*. InProceedings of SIGGRAPH 1996,pp 225-236, ACM Press.

[2] Chun, B. K., Ryu, D.S., Hwang, W.I., and Cho, H. G., *An automated procedure for word balloon placement in cinema comics*. The 2nd International Symposium on Visual Computing, 2006.

[3] McCloud, S. (1999). *Understanding comics*: The invisible art. Kitchen Sink Press

[4] McCloud, S. (2000). *Reinventing comics: How imagination and technology are revolutionizing an art form.* Perennial

[5] W.I. Hang and B. K. Chun and D.S. Ryu and H. G. Cho. (2006) *Cinema Comics: Cartoon generation from video stream.* Proceedings of GRAPP 2006, pp 299-304