

문맥상 자연스러운 멀티동화 자동 애니메이션을 위한 참조현상 해결*

홍경화^o 박종철
한국과학기술원 전산학전공 및 첨단정보기술 연구센터
{kwhong,park}@nlp.kaist.ac.kr

Anaphora Resolution for Contextually Appropriate Animation of Multimedia Fairy Tales

Kyung Wha Hong^o Jong C. Park
Computer Science Division & AITrc, KAIST

요 약

참조현상이란 이미 언급되었던 혹은 이미 알고 있다고 여겨지는 정보에 대한 재표현이다. 참조현상은 자연언어처리 분야에서 뿐만 아니라 인지과학, 심리학, 철학분야에서도 활발하게 연구되는 현상으로 참조표현인 조응사(anaphora)의 선행사(antecedent)를 채택하는 방법에 따라 그 성능이 좌우된다. 자연언어문장으로부터 멀티동화를 생성을 위한 애니메이션 제어 스크립트 명령들에서의 참조해결은 선행 정보의 적절한 참조를 바탕으로 자연스러운 애니메이션 장면을 생성하는데 있어서 필수적이다. 본 논문에서는 이러한 동화의 자연언어 문장에 나타나는 참조현상들에 대해 살펴보고 결합범주문법을 이용하여 참조현상을 해결하는 방법과 구현방법에 대해 논의한다.

1. 서론

자연언어는 형식언어에 비하여 사용자가 원하는 바를 자연스럽게 표현할 수 있다는 장점을 갖는다. 이러한 자연언어는 컴퓨터와 인간사이의 의사전달 수단으로 크게 각광 받고 있는데 본 연구에서는 자연언어로부터 3차원 가상공간의 애니메이션을 생성하는 시스템에 대해 다룬다. 3차원 가상공간의 애니메이션은 2차원 애니메이션에 비해 시공간을 효과적으로 표현하고 각 객체와 효과를 독립적으로 제어할 수 있는 특징을 가진다. 따라서 자연언어 문장으로부터 애니메이션 제어 명령이 적절히 분석된다면 사용자의 표현 의도를 쉽게 애니메이션으로 생성할 수 있게 된다.

[1, 2]에서는 동화의 자연언어 문장을 3차원 애니메이션 제어 명령들로 생성하기 위해서 캐릭터와 그 동작들 사이의 시간 관리, 문장 상에 나타나는 참조해결, 사물의 위치 결정, 캐릭터가 실행할 세부 명령 결정, 다수 캐릭터 제어등에 대한 고려가 필요하다고 밝히고 있다. 본 연구에서는 이러한 고려사항 중에서 특히 참조현상에 대한 해결을 집중적으로 다룬다. 기존의 자연언어 문장상에 나타나는 참조현상은 조응사인 명사구의 선행사를 찾아 해결하고 있다. 그러나 애니메이션 제어 명령을 생성하는 경우에는 객체의 명칭을 나타내는 명사구의 참조해결만으로 자연스러운 애니메이션을 생성하기 어렵다. 그러므로 애니메이션을 알맞게 제어하기 위해서는 동작, 장면, 사건, 시간, 위치 정보등에 대한 참조현상도 같이 고려되어야 한다.

본 연구에서는 이러한 참조현상을 해결하기 위해 결합

* 본 연구는 첨단정보기술 연구센터를 통하여 한국과학재단의 지원을 받았음

범주문법을 사용하여 문장을 분석하고 이를 바탕으로 처리방법을 제안한다. 본 논문의 2절에서는 기존의 참조해결에 관련된 연구들과 자연언어를 애니메이션 제어 명령으로 생성하는 연구들에 대해 살펴보고, 3절에서는 동화상에 나타나는 참조현상들과 그 종류를 살피고, 4절에서는 참조해결 방법을 제안한다. 5절에서는 구현시스템과 참조해결 결과를 다루고 6절에서는 결론 및 향후계획에 대해 논의한다.

2. 관련연구

본 절에서는 기존의 참조현상의 해결에 관련된 연구와 자연언어문장으로부터 애니메이션 제어 명령들을 생성하는 연구들에 대해 살펴본다.

2.1 참조해결

참조현상이란 자연언어처리 분야에서 뿐만 아니라 인지과학, 심리학, 철학분야에서도 활발하게 연구되는 현상으로 이미 언급되었던 혹은 이미 알고 있다고 여겨지는 정보에 대한 재표현이다. 따라서, 참조표현인 조응사(anaphora)의 재표현 대상인 선행사(antecedent)를 제대로 찾아내야만 그 조응사의 의미를 명확히 알 수 있는데 이러한 과정이 참조해결이다.

자연언어처리 분야에서 이루어진 참조해결 연구들에서는 주로 대명사, 명사구, 생략현상을 조응사로 보고 선행하는 모든 명사구를 선행사 후보로 취급한다. 그리고, 선행사 후보들 중에서 주어진 제약 조건과 방법론에 가장 부합하는 선행사를 찾아 참조해결을 하게 된다. [11]에서는 대화문 상에서 나타나는 참조현상에 대해 다루고 있다. 대화문 상에 나타나는 대명사를 조응사로 보고 조응사의 종류를 선행사의 종류에 따라 구분하여 그에 따라 다른 참조해결 방법을 적용하고 있다. 본 연구에서는 동화의 문장을 애니메이션으로 생성하기 위해서 참조현상을 단순히 명사구 뿐만 아니라 동작, 상태, 장면 등의 여러 정보들에 대해서 다루고 있다.

[5]와 [9]에서는 참조현상을 언어현상에서 뿐만 아니라 다른 매체의 특징과 연관하여 다루고 있다. [5]에서는 발화하는 동안 눈의 움직임을 참조해결에 이용하고 있는데 발화전에 눈으로 봤던 곳에 있던 장소나 물체에

해당하는 명사구를 선행사로 결정한다. [9]에서는 마우스와 터치스크린의 정보에 따른 참조해결 방법을 다루고 있는데 마우스나 터치스크린으로 지시되었던 부분을 선행사로 결정한다. 이 분야의 연구에서는 언어 정보 이외에 위치정보, 눈의 움직임, 캐릭터의 행동 등의 다양한 정보가 참조해결을 위해 사용되지만 선행사를 여전히 단순한 표현에 대해서만 다루고 있다.

2.2 자연언어문장의 애니메이션 제어 명령 생성

[14]는 발화된 자연언어 명령을 통해 3차원 캐릭터들을 제어하는 시스템으로 참조현상과 생략현상에 대해 다루고 있다. 이 연구에서는 발화된 자연언어 명령을 정의된 격틀 구조로 표현하여 간단한 제어명령을 생성하고 격틀 간의 비교를 통해서 참조현상과 생략현상을 해결한다. 하지만 본 연구와는 달리 자연언어 문장상에 다양하게 나타나는 다른 정보들간의 참조현상에 대해서는 다루고 있지 않다.

[13]은 자동차 사고 상황을 기술한 자연언어 문장을 3차원 애니메이션으로 생성하는 방법에 대한 연구이다. 이 연구에서는 애니메이션 명령을 생성하기 위한 정보를 문장에서 추출하여 정의된 틀로 표현하고 사고 상황을 재현하는 명령을 생성한다. 애니메이션을 위한 문장이 자동차 사고 상황을 기술한 문장이라는 점에서 동화의 문장들보다 상황이나 사물, 처리 정보 등이 제한적이어서 제어 명령을 생성하기는 수월하다고 볼 수 있다. 하지만 정의된 틀로 표현할 수 있는 상황이 제한적이어서 다양한 상황을 재현하기 힘들고, 참조해결이 필요한 경우나 복잡한 문형에 대한 처리는 다루고 있지 않다.

3. 동화에 나타나는 참조현상

관련 연구에서 논의한 것과 같이 기존의 참조해결은 주로 명사구나 대명사를 대상으로 하고 있다. [6]에서는 명사구 뿐만 아니라 시제, 가정, 동사구 생략, 강세와 같은 다양한 현상도 참조현상으로 고려하는 연구들에 대해 다루고 있다. 본 절에서는 자연언어문장을 애니메이션 제어 스크립트 명령으로 생성하는 경우에 나타나는 참조현상을 살펴보고 그 종류들에 대해 논한다.

3.1. 동화 문장에서의 조응사

동화의 자연언어문장을 애니메이션으로 생성하기 위해서는 문장 상의 참조현상 뿐만 아니라 애니메이션 명령과 장면상에 나타나는 참조현상도 고려되어야 한다. 그러므로 문장에서 애니메이션 제어 명령 요소들인 동작, 주체, 대상이 우선 결정되어야 하고, 각 요소의 성질인 사물, 동물, 배경, 동작, 사건, 시간, 위치 정보가 분석되어야 한다. 이때 생성되는 명령에서 참조해결은 명령 요소가 이미 생성되었던 스크립트 요소이거나 동작이나 상태의 상대적인 변화를 의미하는 경우이다. 이에 따라 본 연구에서는 기존의 명령 요소의 명칭을 나타내는 명사구 뿐만 아니라 각 명칭이나 동작의 성질을 결정하는 부사구나 동작을 의미하는 서술어구도 조응사의 가능성을 가지게 된다고 본다.

3.2. 조응사의 종류

동화의 자연언어 문장을 애니메이션으로 생성하기 위해서는 3.1절에서 논의한 것과 같이 각 명령 요소의 명칭과 성질들이 조응사로 고려되어야 한다. 본 절에서는 동화의 자연언어 문장에서 나타나는 조응사를 큰 의미에 따라 아래와 같이 분류하였고, 예를 바탕으로 참조현상과 관련된 정보들을 분석하였다.

• 객체 (agent, object)

애니메이션 제어 명령 요소 중 주체(agent)와 대상(object)은 객체를 나타내고, 객체의 명칭과 성질 (종류, 동작/상태, 위치, 배경, 시간) 이 조응사가 될 수 있다.

(1) 이곳 저곳을 살피던 여우는 가장 낮은 곳에 있는 포도를 찾아내었습니다. "저거라면 내가 따먹을 수 있겠다."

위 (1)의 첫번째 문장에서 '여우'는 주위를 살피는 시선과 고개를 이동하는 동작을 하고 있고, '포도'는 '가장'이라는 부사의 의미로 셋 이상의 포도 중 가장 낮은 위치의 포도를 지칭하고, 여우가 원하는 포도를 찾아내어 시선이 해당하는 포도를 향하고 있음을 알 수 있다. 두 번째 문장의 '저거'는 객체의 명칭을 지칭하는 조응사로 사물을 가리키는 대명사이므로 앞문장의 포도가 선행사이고, '내'는 동물을 가리키는 조응사로 앞문장의 이곳 저곳을 살피다가 포도를 찾아낸 여우가 선행사가

된다.

• 동작 (action)

애니메이션 제어 명령 요소 중 동작은 동물의 경우는 움직임, 사물의 경우는 상태를 나타내는 요소이다. 동작의 경우에는 그 명칭과 성질(동작/상태, 위치, 배경, 시간)들이 조응사가 될 수 있다.

(2) 여우는 포도를 따먹기 위해 힘껏 뛰어올랐습니다. 여우는 있는 힘을 다해 또 뛰어올랐지만 역시 포도를 떨 수가 없었습니다. 여우는 포도를 먹고 싶은 마음에 하루종일 덩굴 아래서 뛰어오르기를 하였습니다.

(2)의 예문에서 동작에 관련된 참조현상을 살펴보면, 첫번째 문장의 경우에는 여우가 뛰어오르는 동작을 하는데 그 위치가 포도 아래이다. 두 번째 문장의 경우에는 뛰어오르는 동작이 앞문장의 뛰어오르는 동작과 동일하므로 조응사이고, '또'라는 부사에 따라 동작이 한번 반복되고 선행문장의 동작보다 힘이 들어간, 보다 높이 뛰는 동작임을 알 수 있다. 세 번째 문장의 경우에는 선행문장에 나타났던 뛰어오르는 동작을 하는데 시간정보가 '하루종일'로 이 동작이 하루의 단위시간 동안에 걸쳐 반복됨을 알 수 있다.

• 사건

동화의 문장에서는 선행된 애니메이션 장면인 사건을 선행사로 갖는 조응사도 나타난다. 선행사가 사건인 경우에는 (3)과 같이 이미 생성되었던 장면이거나 굳이 장면으로 생성하지 않아도 되는 경우가 대부분이어서 조응사로 분류는 하지만 제어명령으로 분석하지는 않는다.

(3) 한 입, 두 입 퍼먹다 보니까 금새 꿀단지 바닥이 나고 말았어요."어떡하지?훈장님께 꾸중을 들을텐데." 그런데 갑자기 한 아이가 훈장님의 버투를 마당에 내동댕이 쳐어요. 이윽고 훈장님이 돌아오셨어요. "아니, 이게 어찌된 일이나?"

• 특수한 경우

동화의 문장에서는 조응사를 해석하는 주체에 따라 선행사가 달라지는 (4)와 같은 경우도 나타난다. '거짓말쟁이 영감'의 선행사는 도깨비 입장에서는 지금 만난 '욕심쟁이 흑부리 영감'이고 의미상으로는 그 전에 만

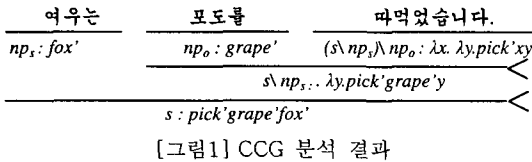
났던 ‘혹부리 영감’이다. 또한 조용사의 의미 정보에 따라 (5)와 같이 여러 개의 선행사를 갖는 경우도 있다.

- (4) 이 소문을 듣고, 욕심쟁이 혹부리 영감도 산으로 가서 노래를 불렀죠. 이번에도 어디선가 도깨비들이 나타났어요." 거짓말쟁이 영감! 잘 만났다. 한번 혼 좀 나 보시지."
- (5) "하하하! 또 속을 줄 알아? 이 흑이나 도로 가져가시지..." 욕심 많은 혹부리 영감은 흑을 떼려 갔다가 싹혹부리가 되어서 돌아왔답니다

위에서와 같이 본 연구에서는 조용사의 종류를 크게 객체, 동작, 사건, 특수한 경우들에 대해 나누었지만 실제로 처리하는데 있어서는 객체와 동작에 관련된 정보들을 고려하여 참조현상을 처리하였다.

4. 동화에 나타나는 참조현상 해결

본 절에서는 멀티동화 애니메이션을 위한 자연언어문장상의 참조현상을 처리하기 위한 방법론을 제시하고 결합범주문법(CCG)을 이용한 처리 방법을 다룬다. 결합범주문법은 범주문법에 결합자가 접목된 문법체계로 어휘범주에 통사정보 외에 의미, 담화정보를 추가하여 [그림 1]과 같이 통사, 의미, 담화 등의 여러 단계 분석을 한 번의 유도과정으로 해결할 수 있다는 특징을 갖는다 [8].



4.1. 선행사 후보

명사구 위주의 참조해결 방법에서는 명사구의 선행사 후보를 선행된 모든 명사구로 보는데 반해 본 연구에서는 이미 나타났던 명령 요소들의 정보인 사물, 동물, 배경, 동작, 사건, 시간과 같은 정보들을 선행사 후보로 본다.

4.2. 선행사 결정

기존의 참조해결 연구에서는 성/수의 일치, 의미 일치, c-command 제약, 통사정보의 일치, 초점, 근접성 등이 조용사의 선행사 후보의 범위를 줄이는 역할을 한다. 본 연구에서 선행사 후보의 범위를 결정하는 요인을 통사정보, 담화정보, 의미정보 (스크립트 명령 요소의 종류, 스크립트 명령 요소의 명칭, 스크립트 명령 요소의 성질), 근접성으로 정하고 선행사 후보의 개수를 줄여나가고 선행사를 결정하는 방법을 제안한다.

4.2.1. 담화 정보

담화에서 화제란 문장이 말하고자 하는 주요 내용을 의미하는데 [7]에서는 문맥 정보를 유지하는 화제의 개념이 참조해결 시에 선행사 후보의 범위를 줄이는 결정적인 역할을 한다고 밝혔다. 화제는 문장의 필수적인 요소로 조용사의 선행사 후보의 범위를 줄여주기 때문에 참조해결 시 필수적인 고려 요인이 된다. 문장의 화제정보는 각 문장마다 유지되며 오래된 화제정보는 선행사 후보에서 탈락하게 된다. [10]에서는 문맥정보의 화제를 유지하기 위해 조용사가 많이 쓰이고 있다고 한다. 따라서 담화의 화제가 이미 선행되어 나타났던 정보의 내용이고 논평이 새로 나타난 정보라고 정의하면, 화제는 선행 정보를 참조해야 하는 조용사의 가능성을 가지고 논평은 선행 정보를 참조할 필요가 없는 새로운 정보라고 볼 수 있다.

이러한 논의를 바탕으로 문장을 애니메이션 제어 스크립트 명령들로 분석할 때에는 각 애니메이션 제어 명령 요소가 이미 생성된 모든 명령 요소를 선행사로 가진다고 본다. 그리고, 위에서 밝힌 화제-논평 개념에 따라 해당 명령요소가 조용사인지 아닌지 여부를 판단하게 된다. 즉, 분석된 문장의 명령 요소 정보가 이미 나왔던 정보 혹은 그와 연관된 정보일 경우에는 조용사로 판단하여 참조해결을 수행하고 그렇지 않은 경우에는 새로운 정보로 추가한다.

4.2.2. 통사 정보

본 연구에서는 애니메이션 명령을 생성하기 위해서 어절 단위로 문장의 통사정보를 분석한다. 제어 명령의 틀을 결정하기 위해서 주어는 주체, 목적어는 대상, 서술어는 동작으로 결정된다. 문장의 통사정보는 어절단

위로 분석되기 때문에 조사의 정보를 담기 위해서 np와 s이외에도 조사의 통사론적 격인 주격, 목적격과 의미론적 격인 처소격, 도구격, 시간격, 상대격, 공동격, 시발, 도달[3]로 구분하여 분석한다.

4.2.3. 의미 정보

애니메이션을 위한 스크립트 명령은 문장의 의미정보에 따라 대부분이 결정된다. 각 어절에 해당하는 의미의 종류, 의미에 따른 애니메이션 스크립트 명령, 스크립트 명령의 성질을 결정하는 정보가 CCG의 의미 범주로 할당된다. 이러한 명령 요소들은 각기 고유의 식별자를 가지고 있고, 조음사와 선행사의 관계가 있는 요소들은 같은 식별자가 부여되어 참조 관계를 형성한다.

각 문장의 어절은 그 의미에 따라서 명령 요소의 종류, 명칭, 성질이 결정되게 된다. 스크립트 명령 요소의 종류는 action, thing, animal, place, action, event, time이 되고, 그 종류에 따라 선행사 후보의 범위가 결정된다. 스크립트 명령 요소의 명칭과 스크립트 명령 요소의 성질 또한 그 의미에 따라 해당하는 애니메이션 스크립트 명령으로 분석되는데, 이때 결정되는 명령의 요소(action, object, agent)의 명칭과 각 명령요소의 성질(action, thing, animal, place, action, shape, event, time)이 명령 요소의 종류와 더불어 선행사의 범위를 결정하는 주된 역할을 한다.

위의 정보를 포함하기 위한 어절의 범주는 (6), (7)과 같은 형태로 정한다.

- (6) np(격, 동작/상태)
 - : [(action명칭, 성질, action 조음사의 종류),
 - (agent명칭, 성질, agent 조음사의 종류),
 - (object 명칭, 성질, object 조음사의 종류)]
- (7) s(동작/상태)
 - : [(action명칭, 성질, action 조음사의 종류),
 - (agent명칭, 성질, agent 조음사의 종류),
 - (object 명칭, 성질, object 조음사의 종류)]

참조해결 단계에서 의미정보는 애니메이션 제어 스크립트 명령을 결정하는 주된 정보로 각 어절단위로 CCG 범주의 의미정보에 스크립트 명령요소의 종류, 명칭, 성질로 할당한다. 이때 명령요소의 명칭은 주로 명사구로 나타나지만 종류와 성질 등은 명사구를 수식하

는 관형사절이나 부사로 나타나기 때문에 수식어구의 의미정보도 명령 요소로써 고려된다. '이곳저곳을 살피던 여우는'과 같은 예문은 동작 주체에 해당하는 어구이다. 이를 의미 범주의 주체(agent)와 관련된 항목들에 대해서만 살펴보면 (8), (9) (10)과 같다. 여우(fox)라는 주체는 이곳 저곳을 살피는 (즉, search (fox, here_there)) 행동을 하는 여우로 분석이 된다.

- (8) lex('이곳저곳을',
 - [np,nonact,loc]:[Agent_n,here_there,Agent_a]).
- (9) lex('살피던',
 - [np,act,_]:[action=search(Agent_n,Agent_f),Agent_a++action]/[np,act,subj]:[Agent_n,Agent_f,Agent_a]\[np,nonact,loc]:[Agent_n,Agent_f,Agent_a]).
- (10) lex('여우는',[np,act,subj]:['fox',Agent_f,animal]).
- (11)[np,act,_]:[fox,action=search(fox,here_there),animal++action]

각 어절의 의미에 따라 명령 요소의 종류는 action, thing, animal, place, action, shape, event, time중 하나로 결정되고, 이미 나타났던 스크립트 명령 요소는 종류에 따라 리스트로 유지된다. (8), (9), (10)의 분석 결과인 (11)을 보면 fox와 search(fox,here_there)은 각각 독립된 명령 요소로 취급되며 각각 animal, action 종류의 조음사 후보가 된다.

의미 정보를 바탕으로 한 참조해결 단계에서는 조음사의 선행사 후보들의 범위를 같은 종류의 명령 요소들에서 이미 생성되었던 스크립트 명령 요소로 좁히게 된다. 명령 요소의 종류에 따라 선행사 후보들이 결정되면 선행사 후보의 범위는 해당 리스트의 후보들 중 명칭 정보와 성질이 일치하는 선행사 후보들로 좁혀지게 된다. 이러한 방법으로 선행사 후보가 결정되면 조음사는 선행사와 같은 식별자로 연결되어 애니메이션 생성시에 자연스러운 장면이 생성되도록 한다.

4.2.4. 근접성 (Recency)

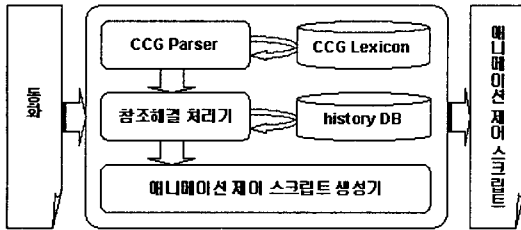
의미 정보에 따른 선행사 후보 결정단계에서 종류, 명칭, 성질에 따른 선행사 후보가 여러 개로 찾아지게 되면 가장 최근에 나타났던 명령 요소가 선행사로 결정된다. 즉, 조음사의 선행사 후보 중에서 다른 성질을 가졌지만 명칭이 같은 선행사가 여러 개 나오면 가장 최근에 생성된 요소가 선행사로 결정된다. 또한 대명사

의 경우에는 명칭은 없고 조용사의 종류만 알 수 있는데 이러한 경우에는 같은 종류의 선행사 후보 중에서 최근에 생성된 명령 요소를 선행사로 결정한다.

5. 구현 및 결과

본 절에서는 4절에서 제시한 방법으로 구현한 시스템의 구조를 살펴보고 실제 동화의 자연언어 문장을 실행한 결과를 다루고 있다. 또한 생성된 애니메이션 제어 명령에 따른 장면 생성의 예를 보인다.

5.1. 시스템 구조



[그림 2] 시스템 구조

시스템 구조는 [그림2]와 같다. CCG 파서에서는 문장의 통사정보, 의미정보가 분석된다. 참조해결 처리기에서는 분석 결과인 애니메이션 제어 명령 요소 정보들을 담화정보, 통사정보, 의미정보, 근접성을 기준으로 4절에서 언급한 방법대로의 참조해결을 수행하여 각 명령 요소에 고유의 식별자를 부여한다. 이때 조용사에 해당하는 명령 요소는 참조해결로 결정된 선행사와 같은 식별자를 가지게 된다. 이러한 결과를 바탕으로 동화를 애니메이션으로 생성하는 제어 스크립트가 생성된다.

5.2. 결과

본 절에서는 논의된 참조해결 방법에 따라 '여우와 신포도'라는 동화의 일부분인 (12)~(16)의 처리 결과들에 대해 다루고 있다.

참조해결 결과인 [그림 3]에서 각 명령 요소의 제일 앞 자리에 있는 숫자가 명령 요소의 고유 식별자이고, 각 식별자에는 명령요소의 명칭과 성질이 저장된다. (12)에서는 주체, 대상, 동작에 해당하는 fox, grape,

search외에 그 성질에 해당하는 search, hang에도 각기 식별자가 부여된다. (12)~(16)에서 나타나는 여우와 포도는 모두 같은 식별자를 갖게 된다. (12)의 포도가 있는 hang이라는 동작은 (16)의 달려있는 hang이라는 동작과 같은 식별자를 갖고 (14)와 (15)의 뛰어오르는 jump라는 동작도 같은 식별자가 부여된다. 문장을 처리하는데 있어서 각 어절의 의미 정보를 모두 표현하지는 않아서 간단한 수준의 참조해결만이 결과로 나와있다. 하지만 각 명령요소 뿐만 아니라 성질들도 선행사의 대상이 되어 참조해결이 됨을 볼 수 있다. 또한, 결과에서는 같은 명칭을 가진 다른 대상을 선행사로 갖는 경우가 보여지지 않지만 명령요소의 성질에 따라 선행사가 결정되기 때문에 해결이 가능하다.

(12) 이곳 저곳을 살피던 여우는 가장 낮은 곳에 있는 포도를 찾아내었습니다.

```
0::fox::action=search(fox,here,there)::
1::search(fox,here,there)::
2::grape::action=hang(grape,low)::num=up_3::
3::hang(grape,low)::
4::find(fox,grape)::
```

(13) 저거라면 내가 따먹을 수 있겠다.

```
0:::
2::that::
5::pick(l,that)::
```

(14) 여우는 포도를 따먹기위해 힘껏 뛰어올랐습니다

```
0::fox::action=pick(fox,grape)::
2::grape::
6::jump(fox,grape)::level=hard::
```

(15) 그렇지만 번번이 입에 닿을 듯 말듯 포도는 떨어지지 않았습니다.

```
0::fox::
2::grape::
6::jump(fox,_)::num=up_1::
7::not_drop(grape,grape)::num=up_1::
```

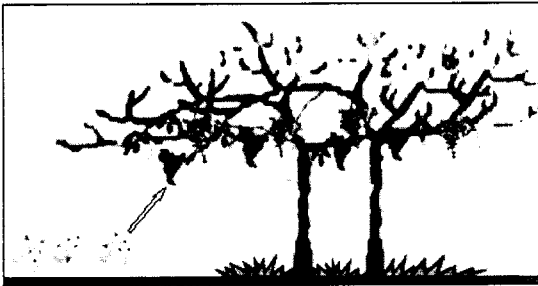
(16) 이 포도는 왜 이렇게 높이 달려있는 거야.

```
0::fox::
2::grape::dist=close::
3::hang(grape,level=high)::level=high::
```

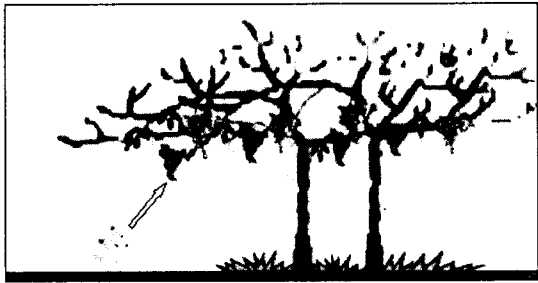
[그림 3] 참조해결 결과

[그림 3]에 따른 애니메이션 제어 스크립트를 기준으로 문장 (12)~(16)에 대한 애니메이션 예상결과는

[그림 4]~[그림 8]과 같다. 여우의 동작과 위치를 기준으로 결과를 살펴보면 [그림 4]는 조금 떨어진 곳에서 포도가 있는 곳을 살펴던 여우가 가장 낮은 곳의 포도를 발견하고 그 포도에 시선을 고정시킨다. [그림 5]는 여우가 발견한 포도에 시선을 고정시키고 발화를 하고 [그림 6]은 그 포도를 향해 뛰어 오른다. [그림 7]은 여우가 여러 번 낮은 곳의 포도를 따먹으려고 하지만 성공하지 못하고 [그림 8]은 그 포도의 바로 아래 위치에서 발화를 하는 장면이다. 이러한 여우의 위치, 시선, 동작은 각 명령 요소들의 참조현상을 해결함으로써 결정이 된다.



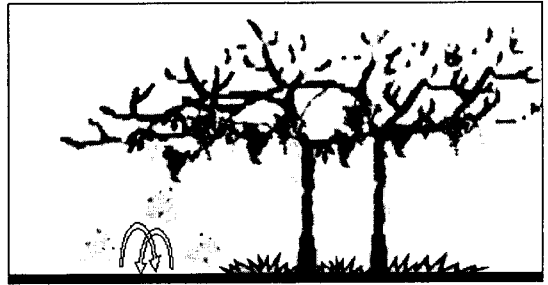
[그림 4]



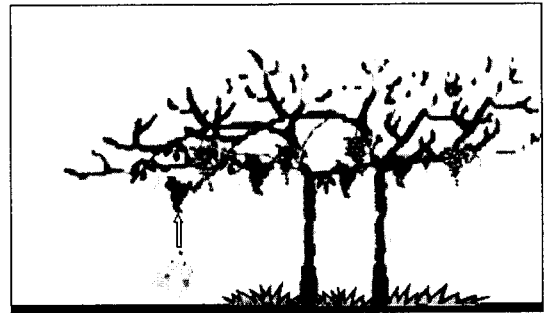
[그림 5]



[그림 6]



[그림 7]



[그림 8]

6. 결론 및 향후 계획

참조표현은 자연언어의 표현능력을 증가시켜 전달효과를 높여주기 때문에 일상적으로 자연언어에서 자주 나타난다. 동화의 문장에서도 이러한 참조표현은 자주 쓰이고 이를 애니메이션으로 생성하기 위해서는 명령 요소들에 대한 참조해결이 필요하다. 본 논문에서는 그러한 참조현상을 기존의 방법과 차이를 두어 살펴보고 그에 따른 해결방법에 대해 논하였다. 구현된 시스템에서는 아직 간단한 의미분석을 통한 처리만을 다루고 있지만 복잡한 명령 요소들의 참조현상도 제안하는 방법으로 처리할 수 있을 것으로 보인다.

참조해결후의 애니메이션 스크립트 명령은 애니메이션 재생기에서 실행시키게 되는데 애니메이션 재생기는 Genesis3D라는 게임엔진을 이용하여 구현 중이다¹. 게임엔진은 상위레벨에서 3차원 가상공간을 제어할 수 있다는 큰 장점을 가지고 있기 때문에 동화의 자연언어

¹ Genesis3D는 공개된 3D 게임엔진으로 <http://www.genesis3d.com>에서 엔진 소스를 제공하고 있다.

문장이 적절한 제어 스크립트로 생성이 되면 정보 표현의 폭을 넓힐 수 있다.

7. 참고 문헌

- [1] 김현숙, 박종철, 결합범주문법을 이용한 실시간 한국어 멀티동화 제작, 한글 및 한국어 정보처리학회 논문집 제 13권, pp. 509-515, 2001.
- [2] 김현숙, 결합범주문법을 이용한 멀티동화 자동 생성, 한국과학기술원 전자전산학과 석사학위논문, 2002.
- [3] 이관규, 조사의 통사론적 연구, In 국어의 격과 조사, 한국어학회, pp. 289-317, 1999.
- [4] Christopher Kennedy, Branimir Bogurev, Anaphora for Everyone: Pronominal Anaphora Resolution without a Parser, COLING, 1996.
- [5] Ellen Campana et al. Using eye movements to determine referents in a spoken dialogue system, In Workshop on Perceptive User Interfaces, ACM Digital Library: November, 2001.
- [6] Emiel Krahmer, Paul Piwek, Varieties of anaphora, ESSLLI, 2000.
- [7] Graeme Hirst, Anaphora in natural language understanding, Springer-Verlag Berlin Heidelberg, 1981.
- [8] Mark Steedman, The Syntactic Process, The MIT Press, 2000.
- [9] Michael Streit, Active and Passive Gestures - Problems with the resolution of Deictic and Elliptic Expressions in a Multimodal System, ACL/EACL Workshop on Referring Phenomena in a Multimedia Context and Their Computational Treatment, 1997.
- [10] Mieke Rats, Topic-Comment Structures in Information Dialogues, Institute for Language Technology and Artificial Intelligence : Research Reports, 1994.
- [11] Miriam Eckert, Michael Strube, Resolving Discourse Deictic Anaphora in Dialogues, EACL, 1999.
- [12] Rulsan Mitkov, Robust pronoun resolution with limited knowledge, Proceedings of the 18th International Conference on Computational Linguistics (COLING'98)/ACL'98 Conference, pp. 869-875, 1998.
- [13] Sylvain Dupuy, Arjan Egges, Vincent Legencre and Pierre Nugues, Generating a 3D Simulation of a Car Accident from a Written Description in Natural Language : the CarSim System, Proceedings of ACL Workshop on Temporal and Spatial Information Processing, pp. 1-8, 2001.
- [14] Yusuke Shinyama, Takenobu Tokunaga, and Hozumi Tanaka, "Kairai" - Software Robots Understanding Natural Language, In the 3rd International Workshop on Human-Computer Conversation, 2000.