

텍스트 입력 기반 지화 및 수화 애니메이션 자동 생성에 관한 연구

이금용**

**영산대학교 정보통신공학부

e-mail : office@java-tech.com

A Study on Auto-Generation of Dactylology and Chirolgy Animation from Text Inputs

Geum-Yong Lee**

**Dept. of Information and Communication, Young-San University

요 약

Unicode 와 지화, 수화의 공통점은 각국 언어의 자모 혹은 단어에 고유한 표현양식이 1:1로 대응되어 있다는 것이다. Unicode 의 경우 각 자모별 고유의 헥사코드가 지정되어 있고 지화, 수화의 경우 각 자모별, 단어별로 고유한 동작을 표현하는 손동작이 지정되어 있는 것이다. 본 논문에서는 텍스트 입력에 대응하는 지화, 수화 손동작 그림을 연속적으로 렌더링함으로써 애니메이션 효과를 낼 수 있는 알고리즘과 그 구현에 관한 연구를 소개한다.

1. 서론

전세계로 연결되는 컴퓨터 네트워크, 위성을 기반으로 하는 다채널 다국적 방송과 등 일반인에 대한 정보제공 내용과 형식이 급속도로 진행되고 있는 현재에도, 청각 장애인에 대한 정보제공은 정상인에 비교하여 볼 때 시간적 지연 혹은 내용의 제한, 추가적인 비용 부담이라는 불편함 속에서 진행될 수 밖에 없다.

본 연구는 텍스트 정보를 자동으로 청각장애인을 위한 그래픽 애니메이션 (指話 및 手話를 구성할 수 있도록 하는 그래픽 동화상)으로 변환하여 줌으로써 음성정보(공중파 방송, 인터넷 등)의 수신자간 시간지연, 내용제한, 추가비용 부담을 제거하는 동시에 청각 장애인에 대한 정보화 복지수단을 강구할 필요성에 착안하였다.

지화 애니메이션 생성을 위해서는 텍스트를 연속되는 유니코드[1]의 스트림 (Stream)으로 가정한다. 예를 들어, “정보처리”라고 하는 텍스트는 “3148 3153 3147 3142 3157 314A 3153 3139 3163”과 동일한 것이다. 각 자모에 대한 고유의 손동작이 정의되어 있으므로, 기본적으로는 해당 유니코드에 대응하는 손동작 이미지 파일을 일정한 속도로 연속하여 직접 렌더링하는 방법으로 애니메이션을 생성할 수 있다.

수화의 경우에는 약간의 전처리 작업이 필요하다. 예를 들어, “가로채다”라는 말은 “오른손을 안으로 끌어 들이며 주먹을 쥔다” 라는 식으로 정의되어 있어서, 이에 해당하는 애니메이션 동화상을 미리 제작하여 저장해둘 필요가 있는 것이다. 입력되는 텍스트는 일단 해당 수화단어를 나타내기 위해 특별히 고안된 zip 코드의 스트림으로 변환된 후, 사전 저장된 동화상 디렉터리에서 zip 코드 순서대로 애니메이션을 합성해야 한다. 물론, 助詞등을 위한 지화 손동작 등을 보조적으로 사용할 수 있다.

본 논문에서는 텍스트를 지화 및 수화 애니메이션으로 자동 변환하기 위한 알고리즘과 그 구현 방법에 대하여 소개한다. 제 2 절에서는 지화 애니메이션 자동생성을, 제 3 절에서는 수화 애니메이션 자동생성에 관하여 논한다.

2. 지화 애니메이션

2.1 유니코드 스트림

텍스트는 연속되는 유니코드 스트림으로 변환할 수 있다. 각 유니코드에 대하여 해당 국가의 언어별로 정의된 지화 (Dactylology) 손동작을 대응시켜 두면, 결국 유니코드 스트림으로 변환된 텍스트는 손동작의

연속 이미지로 처리할 수 있다. 그림 1. 은 텍스트를 유니코드 스트림으로 변환하기 위한 알고리즘이다.

그림 1. 에서, Typewriter 는 지화 및 수화 애니메이션으로 변환하기 위한 텍스트를 직접 키보드에서 입력하거나, 해당 텍스트 파일을 준비하는 사람이다. Character Parser 는 Typewriter 에 의해 입력되는 텍스트의 각 자모를 인식하여 해당 자모에 대응하는 언어별 Unicode 값을 반환(return)하는 프로그램이며, Character Sequencer 는 Character Parser 에 의해 반환되는 Unicode 값을 순서대로 Character Sequence 에 저장하는 프로그램이다. Character Sequence 는 Typewriter 의 입력 텍스트 각 자모에 대응하는 Unicode 값을 입력순서대로

로 저장하고 있는 오브젝트이다. 즉, 입력 텍스트가 유니코드 스트림으로 변환된 결과를 저장한다. 벡터, 별도 저장 파일, 데이터베이스 등을 사용할 수 있겠다.

키보드에서 직접 입력되지 않고, 미리 입력되어 저장된 텍스트 파일의 경우에도 그림 1.의 알고리즘이 그대로 적용된다. Java 의 경우, 해당 파일에 대하여 DataInputStream 을 연결한 후 readUTF 메서드를 사용하면 텍스트를 유니코드 형식으로 읽어들이 수 있다. 이후에는 그림 1.의 과정에 따르면 된다. 개발된 소프트웨어의 경우, 키보드로부터 입력을 받을 것인가 혹은 파일로부터 입력을 받을 것인가 선택할 수 있도록 되어 있다.

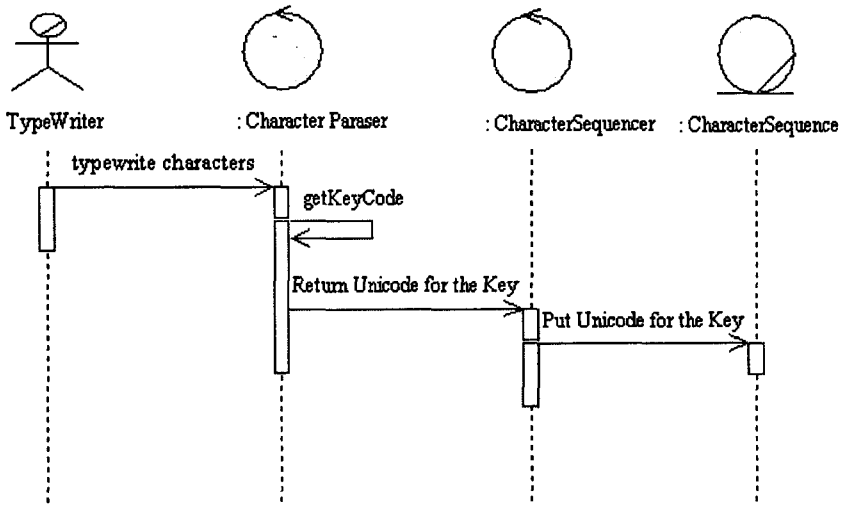


그림 1. 텍스트 입력을 유니코드 스트림으로 변환하기

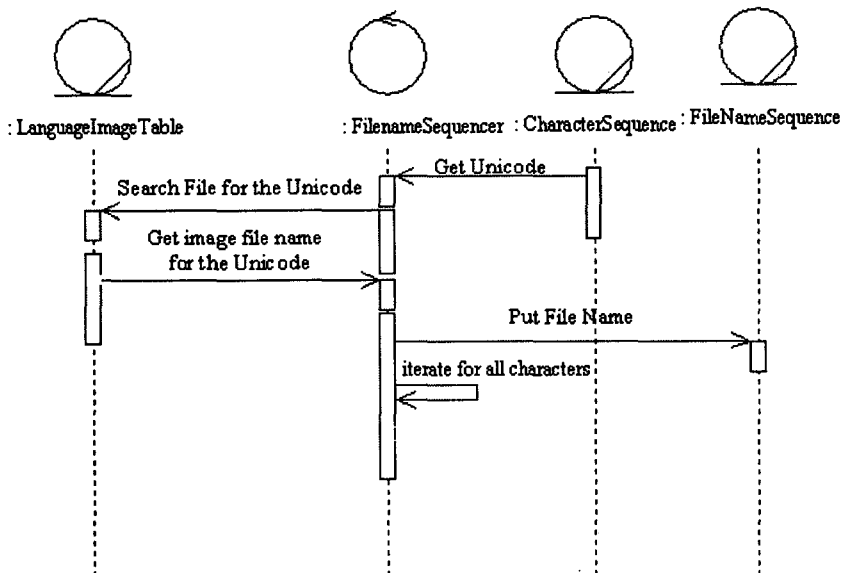


그림 2. 유니코드 스트림을 이미지 파일명 스트림으로 변환하기

2.2 지화 이미지 파일명 스트림

유니코드 스트림으로 변환된 입력 텍스트는 각 유니코드에 대응하는 지화 이미지를 저장하는 파일명 스트림으로 변환되어야 한다. 그림 2.는 이 과정을 보여주고 있다. 각국 언어별로 각 자모에 해당하는 유니코드를 키(Key)로 하고, 해당 이미지 파일명을 값(Value)로 하는 테이블인 LanguageImageTable 을 우선 준비해야 한다. Java 의 Hashtable 오브젝트를 사용하면 될 것이다. FilenameSequencer 는 Character Sequence 에 저장된 Unicode 각각에 대응하는 이미지 파일 목록 FileNameSequence 를 생성하는 프로그램이다.

2.3 지화 애니메이션의 생성

그림 3 은 파일명 스트림 FileNameSequence 에 저장된 이미지 파일을 순서대로 렌더링함으로써 애니메이션을 생성하기 위한 과정을 보여주고 있다. ImageFileDirectory 는 실제 이미지 오브젝트를 저장하는 디렉터리이며, Animator 는 이미지 오브젝트를 PaintingScreen 에 렌더링하는 프로그램이다. Timer 는 렌더링 속도 FPS (Frame per Second)를 제어하며, Animator 와 일체화된 클래스이다.

그림 4.는 지화 애니메이션 생성을 위한 전체과정을 보여주는 알고리즘으로서, 그림 1 ~ 그림 3 을 요약하고 있다.

3. 수화 애니메이션

指話가 각 자모에 1:1 로 대응하는 손동작을 가지고 있는데 반하여, 手話는 개념 혹은 단어에 대하여 연속 손동작이 정의되어 있다. 예를 들면, “빼다 (subtract, subtraction)” 가, “가슴을 향한 왼손바닥에 오른손등을 대고 스쳐 내린다”로 정의되며, “부유하다 (rich)”는 “오른손의 엄지·검지 동그라미를 왼쪽 어깨에 대고 엄지·검지를 편다”로 정의되어 있는 것이다[2]. 정의되어 있지 않는 경우에는 지화를 사용하면 될 것이다. 자모별 구분 지화동작은 제 2 절에서 소개된 내용을 참조하자.

3.1 수화단어 ZIP 코드 스트림

단순한 지화 이미지의 연속동작으로서 정의되지 않는 수화단어에 대하여는 고유한 zip 코드 (본 연구에서는 이를 Chirozip¹이라 함)를 부여할 수 있다. 예를 들어, “부유하다”는 “#cz3633” 이 될 수 있다. 여기에서 “#cz” 는 유니코드가 아닌 Chirozip 코드로 해석하기 위한 Escape Character 이다. 각국 언어별로 해당 단어에 대한 Chirozip 코드 디렉터리를 제작하고 표준화할 수 있을 것이다.

애니메이션 변환을 목적으로 텍스트를 입력할 경우, Chirozip 에 의한 해석이 필요한 경우, “[부유하다]” 와 같이 특수기호와 함께 입력하면, 그림 1.의 Character Parser 및 Sequencer 는 제 2 절에서 소개한

유니코드 대신에 “#cz” 를 삽입한 Chirozip 코드를 입력하게 된다.

따라서, 주어진 텍스트는 유니코드와 Chirozip 코드의 혼합 형태로 변환되어, 그림 1 의 Character Sequence 에 저장된다.

3.2 수화단어 동영상 파일명 스트림

Character Sequence 에 저장된 유니코드와 Chirozip 코드는 그림 2.의 FilenameSequencer 에 의해 유니코드와 Chirozip 코드에 대응하는 파일명 스트림으로 변환되어 FileNameSequence 에 저장된다.

그림 2.의 LanguageImageTable 에는 각국 언어별 유니코드 이외에 Chirozip 코드를 Key 로 하는 동영상 파일명 Value 가 추가된다.

3.3 수화 애니메이션의 생성

그림 3.의 Animator 는 Chirozip 코드에 해당하는 동영상 파일을 유니코드 이미지 파일의 전후에 삽입, 렌더링함으로써 수화 애니메이션을 생성한다. 물론, 사전 제작되어 저장된 동영상 파일의 FPS 와 유니코드 이미지에 대한 렌더링 FPS 는 동일하게 설정된다.

4. 결론

본 논문에서는 텍스트 입력으로부터 지화 및 수화 애니메이션을 자동 생성하기 위한 알고리즘을 소개하였다. 각국 언어별로 자모 대응 지화 손동작이 1:1 로 정의되어 있고, 추상적인 개념에 대한 수화단어 연속 동작 또한 1:1 로 정의되어 있다는 점에 착안하여, 프로그램에 의해 사전 저장된 이미지 오브젝트 및 동영상 파일을 연속으로 렌더링하는 것이 핵심이다.

지화 애니메이션의 경우 유니코드를 이미지 오브젝트와 대응시키고, 수화 애니메이션의 경우 특정 수화 단어에 대응하는 Chirozip 코드를 개발하고 이를 표현하는 사전 제작 동영상과 대응시키면 되는 것이다.

유니코드 대응 이미지 파일과 Chirozip 코드 대응 동영상 파일은 다양한 제작 툴을 이용하여 제작하면 될 것이다. 다만, 각국 언어별 수화사전에 등록된 수화단어에 대하여 Chirozip 코드를 표준적으로 규정하는 일은 향후의 연구과제로 남긴다.

뉴스, 스포츠, 드라마, 오락물 등 공중과 방송물에 대하여 청각 장애인용의 애니메이션 프리젠테이션이 가능하고, 각종 유무선 정보화 장치를 통한 통신 콘텐츠의 애니메이션 프리젠테이션, 컨퍼런스 내용의 실시간 애니메이션 프리젠테이션 등에 활용될 수 있을 것으로 기대된다. 자연어 처리기술이 필요한 음성인식에 대한 논의는 본 연구의 범위에서 벗어나지만, 속기사에 의한 현장 입력이라는 방법을 사용하면 될 것이다.

참고문헌

- [1] 유니코드 콘소시엄 (<http://www.unicode.org>)
- [2] 한국청각장애인 정보센터, “수화 사전”, 2001

¹ 수화를 나타내는 Chirology 와 ZIP 의 합성어

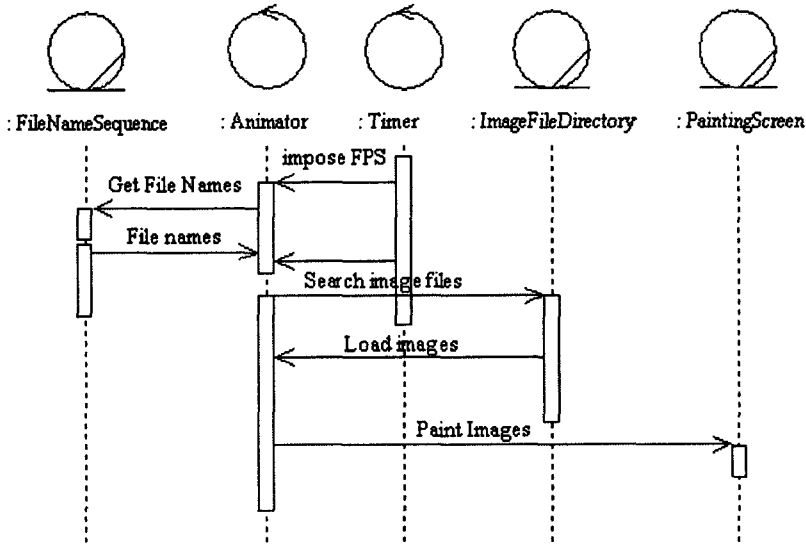


그림 3. 이미지 파일명 스트림으로부터 애니메이션 생성하기
Text Inputs(Keyboard, File)

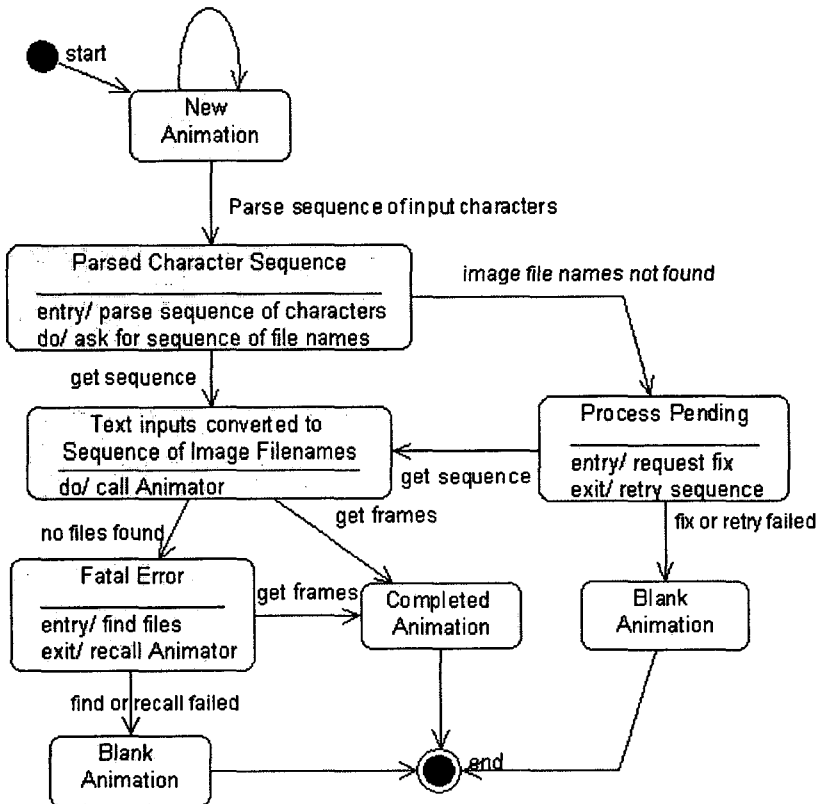


그림 4. 텍스트 기반 지화 및 수화 애니메이션 생성 과정요약