

구조문서 기반 온라인 교정 시스템의 설계 및 구현

정한상*, 김재경*, 손원성*, 임순범**, 최윤철*

*연세대학교 컴퓨터과학과

**숙명여자대학교 멀티미디어학과

e-mail : hs151@rainbow.yonsei.ac.kr, ki187cm@rainbow.yonsei.ac.kr,

sohnws@rainbow.yonsei.ac.kr, sblim@sookmyung.ac.kr, ycchoy@rainbow.yonsei.ac.kr

Implementation of Online Editing System based on Structural Documents

Han-Sang Jung*, Jae-Kyung Kim*, Won-Sung Sohn*, Soon-Bum Lim**, Yoon-Chul Choy*

*Dept. of Computer Science, Yonsei University

**Dept. of Multimedia, Sookmyung Women's University

요 약

최근 웹을 기반으로 한 문서의 전자화가 이루어지면서 기존의 전통적인 펜기반 교정 시스템 또한 온라인 상의 전자 문서 환경에 맞게 변화하고 있다. 이러한 펜기반 입력 기법을 사용하는 교정 시스템에서는 일반 문서와 달리 웹 문서의 구조정보를 고려한 편집이 지원되어야 하며 또한 교정 부호와 텍스트 간의 정확한 영역 인식이 이루어져야 한다. 본 연구에서는 온라인 교정 시스템 모델링을 통하여 온라인 환경에 적합한 교정 부호를 정의하고, 교정 대상 텍스트 영역을 편집 가능한 단위로 구분하여 효율적인 편집 연산이 이루어 질 수 있도록 하였다. 또한 웹 기반의 구조문서(HTML/XML) 편집 환경을 고려하여 편집으로 인한 문서의 구조 정보 변경을 지원하기 위하여 텍스트를 비구조 및 구조정보 텍스트로 분류하여 정의하였다. 본 연구에서는 이러한 모델에 기반하여 교정 부호의 특성에 따른 가변적인 편집 텍스트 영역 인식 규칙 모델을 정의하여 교정 부호와 편집 텍스트 영역간의 모호성을 최소화 하고, 편집으로 인한 문서의 구조 정보 변경을 지원하는 시스템을 구현하였다. 결과적으로 온라인 웹 문서 환경에서 펜기반의 모호한 교정 부호의 입력을 인지적인 관점에서 해석하여 보다 정확한 교정 작업 수행을 지원하도록 하였다.

1. 서론

최근 기존의 종이 환경 문서들의 전자 문서화가 빠르게 진행되면서 종이 문서 환경과 관련된 여러 기법들도 디지털 환경에 맞게 변화하고 있다. 교정 시스템도 이러한 기법중의 하나로, 기존에는 종이 문서 환경에서 교정자가 펜을 이용하여 교정부호를 기입하고 원본 문서 작성자가 교정 부호를 보고 문서를 수동으로 수정하는 방식을 사용하여 왔다. 그러나 모든 문서의 디지털화가 이루어지면서 교정 방식도 기존 방식에서 좀 더 자동화된 오프라인 교정 환경으로 발전하였고 최근에는 온라인 환경에서 교정자와 원본문서 작성자간에 직접 전자 문서를 대상으로 한 교정이 이루어 지고 있다 [1].

그러므로 온라인 상에서의 교정 기법에서는 기존 종이 문서 환경에서 사용되었던 교정 방식과 가장 유사한 기법을 제공해 주어 가장 효율적이고 거부감이 없는 환경을 구축하는 것이 중요하다[5]. 또한 전자 문서의 특성을 고려하여 기존

환경에서 가능하지 않았던 다중사용자간의 협업작업이나 교정 절차의 자동화 등을 지원하는 것이 바람직하다. 특히 전적으로 펜을 이용한 부호 입력 방식으로 이루어지던 교정 절차를 온라인 환경에서도 유지하기 위해서는, 온라인 문서에서의 정교한 텍스트 편집을 위한 펜 기반 입력 기법이 연구되어야 하며, HTML 이나 XML 과 같은 온라인 문서 포맷의 특성인 구조를 고려한 편집 기법 또한 고려되어야 한다.

기존 연구에서는 이러한 점을 고려하여 펜 기반의 교정 시스템들을 연구하였으나 단순히 원본 문서의 텍스트 영역과 입력된 교정부호의 영역만을 비교하거나, 문서의 구조 정보를 고려하지 않기 때문에 현재 웹 문서에 대한 정교한 텍스트 편집 환경을 지원하기에는 부족함이 있다.

본 논문에서는 HTML/XML 기반의 웹 문서환경에서 펜기반의 교정 기법을 이용하여 정확한 텍스트 편집 및 구조 정보를 고려한 편집 기능을 지원하려 한다. 이를 위하여 온라인 문서에서 입력되는 교정 부호와 문서간의 관계를 분석하

고 교정 부호의 특성에 따라 가변적인 텍스트 영역을 인식한다. 또한 구조 문서의 편집을 위하여 명시적 및 묵시적인 구조 정보 변경한다. 이를 위하여 본 논문에서는 규칙 기반의 텍스트 인식 및 구조 정보 변경 기법을 제안한다.

2. 관련 연구

Farkas & Poltrock[1]에 따라 온라인 교정 모델은 목시적 편집, 코멘트, 편집 추적, 전통적 모델로 분류할 수 있다. 여기서 전통적 모델은 기존의 종이 환경에서 쓰던 교정 방식을 그대로 채용한 것으로서 표준적인 교정 부호를 그대로 사용한다. 따라서 저자와 교정자에게 익숙한 교정 환경을 제공하고 정교한 펜기반 입력 기법이 요구되는 모델이다.

이러한 전통적 모델을 채택한 기존 시스템에는 MATE[2], PenEdit[2], 그리고 Amaya+PEN[3] 등이 있다. MATE 와 PenEdit 는 펜기반 입력 및 제스처 인식을 기반으로 하며 원본 문서 및 교정부호간의 네이게이션 기법도 지원한다. 또한 비동기적인 다중 사용자 지원(MATE)을 하며 음성 명령(PenEdit)도 가능하다.

그러나 이 시스템들은 교정 부호 입력 및 인식에 있어서 교정이라는 환경을 고려하지 않은 일반적인 제스처 및 텍스트 인식 기법을 사용하고 있으며 문서의 환경 또한 단순 텍스트 환경이다.

Amaya+PEN 은 W3C 에서 제공하는 HTML 편집기인 Amaya[4]를 기반으로 한 교정 시스템으로 펜기반의 웹문서의 편집이 가능하다. 펜기반의 제스처 인식에서는 교정 환경의 특성을 고려한 기하학적 알고리즘을 사용하였으며 텍스트 정보와 부호간의 관계를 고려한 교정 부호 인식 기법을 적용하였다. Amaya+PEN 은 HTML 문서를 기반으로 하기 때문에 편집으로 인한 구조 정보의 변경이 일어날 수 있는데 이것은 Amaya 편집기에서 기본적으로 제공되는 규칙에 따라 처리된다.

위에 언급한 시스템들은 펜기반 입력 방식이 가지는 본질적인 문제적인 모호성을 고려하지 않았다. 즉, 펜으로 입력된 부호의 영역과 대상 텍스트의 영역이 명확하게 일치하기 힘들기 때문에 이러한 모호성을 최대한 줄여야 한다. 또한 교정 시스템에서는 부호의 종류에 따라 텍스트 편집 영역이 글자, 단어, 절 등으로 구분되어지므로 이러한 특성을 고려하여 텍스트 영역의 인식이 이루어져야 한다.

3. 교정 시스템 모델링

교정 시스템 환경에 따라서 교정 시스템 모델이 달라질 수 있다. 본 연구에서는 온라인 환경의 교정 특성을 고려하여 구조 문서를 편집 할 수 있도록 교정 모델을 정의하였다. 즉, 교정 부호의 특성에 따라 텍스트 영역을 문자, 단어, 절 영역 등으로 인식 및 보정하도록 한다. 또한 편집으로 인해 구조 정보의 변경이 요구될 때에 이를 어떻게 처리할 것인지에 대한 정의를 하였다.

1) 교정 시스템은 교정 부호(M), 텍스트(T)와 실행(E)으로 구성된다. 즉, 교정은 텍스트에 대한 교정 부호의 실행으로 이루어진다.

$$C = E(M, T)$$

2) M 은 미국의 교정 부호 표준인 Chicago Style Proofreader's Marks 의 간소화된 집합이다. 추가적으로 온라인 환경의 구조 문서에 필요한 부호들을 포함한다.

$$M = EDIT \cup POSITION \cup STYLE \cup ANNOTATION$$

$$EDIT = \{DELETE, REPLACE, INSERT\}$$

$$POSITION = \{CLOSE_UP, TRANSPOSE, SPLIT_PARAGRAPH, JOIN_PARAGRAPH, MOVE_LEFT, MOVE_RIGHT\}$$

$$STYLE = \{COMPSE, CREATE_LINK, DELETE_LINK\}$$

$$STRUCTURE = \{CREATE_STRUCTURE, DELETE_STRUCTURE\}$$

$$ANNOTATIONS = \{ANNOTATION\}$$

3) T 는 교정 대상이 되는 텍스트로서 비구조적 텍스트(NST)와 구조적 텍스트(ST)로 구분된다. 비구조적 텍스트는 글자, 단어, 문장 및 문단으로 구성되며 구조적 텍스트는 각각의 비구조적 텍스트에 대한 계층적 구조 정보 경로들(XPath)의 집합으로 이루어진다.

$$T = \{NST, ST\}$$

$$NST = \{LETTER, WORD, SENTENCE, PARAGRAPH\}$$

$$ST = \sum_{NST} XPath$$

$$LETTER = \cup \{A..Z | a..z | 0..9 | !..?\}$$

$$WORD = \cup \{LETTER\}$$

$$SENTENCE = \cup \{LETTER, WORD\}$$

$$PARAGRAPH = \cup \{LETTER, WORD, SENTENCE\}$$

4) E 는 텍스트에 입력된 교정 마크를 실행하는 것으로 교정 대상이 되는 텍스트 영역의 인식 연산(R)과 구조 정보의 변경에 대한 규칙(SC)으로 구성되어 있다. 인식 연산 R 은 교정 부호 M 의 각 타입에 따라 선택적으로 텍스트 T 의 교정 영역을 인식하며 구조 정보의 변경은 문서 타입 선언 정의(DTD)에 따라 묵시적 혹은 명시적으로 이루어진다.

즉 묵시적 구조 변경은 교정 시스템의 결정에 따라 문서의 구조를 자동으로 변경하는 것이며 명시적 구조 변경은 문서 구조 변경에 대하여 사용자의 의사 결정이 요구될 때 수동적으로 수행된다.

$$E = \{M, T | SC(R(M, T))\}$$

$$R = \{(Each\ Type\ of\ M) \cap T\}$$

$$SC = \{IMPLICIT(R), EXPLICIT(USER_DECISION)\}$$

4. 텍스트 영역 인식 및 구조 정보 편집

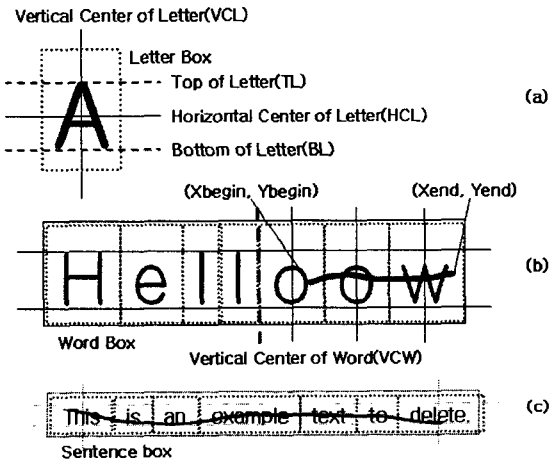
4.1 텍스트 영역 인식

본 장에서는 온라인 교정 부호와 편집 대상 텍스트간에 편집 영역 판별시 발생하는 모호성을 최소화하기 위한 인식 기법에 대하여 설명한다. 제안 기법은 IF-THEN 문법으로 표현되는 규칙 모델을 통하여 처리된다. 본 논문의 규칙 시스템은 총 14 개의 교정 부호 타입과 편집 텍스트 영역 인식을 위한 조건으로 이루어진 90 여개의 규칙으로 정의되어 있다. 본 장에서는 이러한 규칙들 중 텍스트 영역 및 교정 부호 인식과정에 대한 설명을 위해 DELETE 연산의 경우를 예를 들도록 한다.

먼저 입력된 펜기반 제스처는 기존 인식 알고리즘[6]을 이용하여 3 장의 모델링 정의 2 번에 의해 타입이 결정된다. 예를 들어 본 연구에서는 텍스트의 중간을 가로로 긋는 선을 DELETE 라고 정의 하였다. 따라서 [그림 1]의 (b), (c)와 같은 경우 교정 부호의 타입은 DELETE 가 된다.

다음으로 교정 부호가 적용되어질 텍스트 영역을 인식하기

위해서 부호와 텍스트에 몇 가지 기준선 및 점들을 정의하였다. 먼저 모델 정의 3 과 같이 텍스트는 LETTER, WORD, SENTENCE, PARAGRAPH 단위로 구성되며 각 단위는 BOX 로 구분한다. 이 단위들은 [그림 1]과 같이 부호의 시작점 (X_{begin} , Y_{begin})과 끝점(X_{end} , Y_{end}), LETTER 의 중앙수직선(VCL) 와 중앙수평선(HCL), LETTER 의 상단선(TL)과 하단(BL), 단어의 중앙수직선(VCW)등으로 기준선에 의해 영역이 세분화 된다.



[그림 1] 텍스트 영역 기준선 및 DELETE 부호 입력 예

[그림 1]의 (b)와 같은 경우 [표 1]의 규칙에 의해 LETTER 단위의 DELETE 연산으로 판단이 되며 유효한 삭제 대상 텍스트 영역은 마지막의 'ow' 가 된다. 먼저 입력된 교정 부호의 타입이 DELETE 로 인식되고 Y_{begin} 과 Y_{end} 이 TL 과 BL 사이에 포함될 경우 해당 텍스트에 대한 삭제연산이다. 삭제 대상 텍스트 영역을 판별하기 위해 먼저 X_{begin} 과 X_{end} 의 위치가 VCL 의 좌우측에 존재하는가의 여부에 따라 그 위치의 LETTER 를 편집 대상 영역으로 넣을 것인지 결정한다. 위 예에서 5 번째 LETTER 인 'o' 는 교정 부호의 X_{begin} 이 VCL 좌측에 존재하므로 편집 대상 텍스트 영역은 6 번째 'o' 가 된다. 마찬가지로 X_{end} 에서의 편집 대상 텍스트는 'w' 가 된다.

IF statements THEN:actions

- statements : (1) 교정부호 타입이 DELETE 이다
 (2) Y_{begin} 과 Y_{end} 사이의 좌표점들은 TL 와 BL 영역에 포함된다
 (3) 교정 부호의 X_{begin} 및 X_{end} 가 각각 LETTER 의 VCL 좌측 및 우측에 위치 한다
 (4) X_{begin} 과 X_{end} 사이의 LETTER 의 수는 WORD 의 총 LETTER-1 이하이다.
- actions: (1) LETTER BOX 단위의 DELETE 연산이다
 (2) 유효한 LETTER BOX 영역을 추출한다

[표 1]DELETE 인식 규칙 2 번

다음으로 X_{begin} 과 X_{end} 사이의 LETTER 의 개수가 WORD 를

구성하고 있는 전체 LETTER 의 개수보다 작아야 한다. 왜냐 하면 WORD 영역 내에 포함되는 교정 부호는 LETTER 단위의 편집 명령이며 그렇지 않은 것은 WORD 단위 이상의 편집 명령이기 때문이다. 이 예에서는 이 조건을 만족하여 LETTER 단위의 DELETE 명령을 실행하게 되며 만약 그렇지 않다면 WORD 단위의 연산이 되어 다른 규칙으로 처리가 된다.

또한 본 연구에서는 동일 타입의 부호 외에도 각 교정 부호의 타입별 특성에 따라 인식 규칙을 달리 하는 것이 바람직하기 때문에 이를 위한 규칙 모델을 정의하였다. 예를 들어 SPLIT/ JOIN_PARAGRAPH 와 같은 부호는 주로 절단위로 연산이 이루어지기 때문에 이에 맞는 절단위 인식 규칙을 적용하였다.

4.2. 구조 정보 변경 규칙

제안된 구조 정보 편집 규칙모델에서는 입력된 교정 부호와 구조정보와의 관계를 파악하여 구조 정보의 변경이 예측될 시에는 변경 가능한 모든 경우의 수를 산출한다. 예를 들어 사용자가 구조 정보를 포함하는 텍스트에 삭제 연산을 적용하였을 경우 텍스트가 없는 빈 구조정보만을 남겨두거나 텍스트와 함께 구조정보 자체를 삭제 할 수 있다.

본 연구에서는 구조 문서의 DTD 에 위배되지 않는 범위내에서 규칙 모델에 따라 적합한 구조 변경을 묵시적으로 시도한다. 그러나 경우에 따라서 이것은 사용자가 원하지 않는 결과가 될 수 있으므로 가능한 구조 변경 결과들을 추천하여 사용자가 명시적으로 선택할 수 있는 기법을 제안한다.

먼저 묵시적인 구조 변경의 경우는 사용자의 편집 명령으로 인한 문서의 구조 정보 변경의 결과를 정확히 예측 가능할 때 적용될 수 있다. 즉 편집 명령으로 인하여 나올 수 있는 문서 구조 변경의 경우의 수가 한가지 일 때를 말한다. 예를 들어 교정자가 인라인 태그로 묶여 있는 단어를 포함하는 한 문장을 DELETE 명령으로 삭제하였을 때 그 문장이 포함하고 있던 인라인 태그의 구조정보는 문장과 함께 제거된다. 단, 구조 정보 편집 결과가 DTD 에서 허용하지 않는다면 텍스트 노드 없이 빈 태그만을 남겨놓는다.

이와 같은 편집 수행 절차는 다음 [표 3]과 같은 규칙에 의해서 이루어진다.

IF statements THEN:actions

- statements : (1) 교정부호 타입이 DELETE 이다
 (2) T_{begin} 은 1 개 이상의 구조정보를 포함한다
 (3) 포함된 구조정보는 DTD 에서 필수 요소로 정의되어 있지 않다
- actions: (1) 구조 정보의 제거 연산이 발생한다
 (2) T_{begin} 의 텍스트와 포함된 구조정보를 제거한다.

[표 3]DELETE 에 의한 구조정보 변경 규칙 1

교정 부호의 특성이나 구조정보의 위치에 따라 편집 명령으로 변경된 구조정보의 형태가 다수가 될 수 있다. 이러한 경우에 시스템에서 묵시적으로 판단하여 가장 적절한 결과를 교정자에게 출력하여 줄 수 있지만 경우에 따라서는 교정자가 원하지 않는 결과가 나올 수 있다.

따라서 가능한 결과들을 교정자가 화면으로 확인하면서 원하는 결과를 선택할 수 있도록 하는 것이 바람직하다. 이를 위해 가능한 구조정보 변경 결과를 편집 작업시 일어나는 모든 경우에 따라 정의해 놓아야 하며 임의의 편집 명령으로 특정 결과들이 예측될 때는 이를 교정자에게 제공하여 준다. 본 연구에서는 교정 부호 입력에 따른 구조정보 변경 결과들을 32 개의 규칙에 따라 정의하였으며 교정자는 마우스 휠을 회전시켜 시스템에서 추천하는 결과들을 미리보고 원하는 결과를 선택할 수 있다

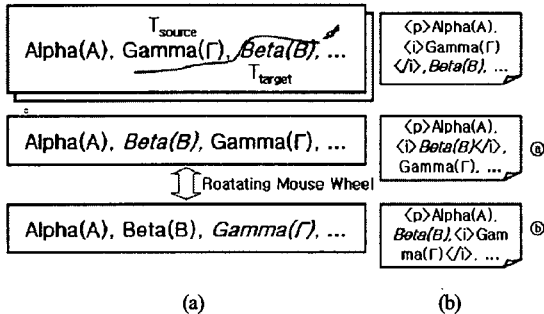
IF statements THEN actions

- statements : (1) 교정부호 타입이 TRANSPOSE 이다
 (2) T_{source} 와 T_{target} 은 각각 단일 구조정보를 가진다
 (3) T_{source} 혹은 T_{target} 영역이 서로 다른 구조정보를 가진다

- actions: (1) 서로 다른 구조 정보에 속하는 TEXT의 교환이 발생한다
 (2) ㉠ 구조 정보를 유지한 채 최하위 노드인 TEXT 만 이동한다
 or
 ㉡ DTD 에 유효하다면 텍스트와 함께 구조정보도 같이 이동한다.

[표 4] TRANSPOSE 에 의한 구조정보 변경 규칙 1

예를 들어 [그림 2]의 (a)와 같은 경우 T_{source} 와 T_{target} 를 교환하는 편집 부호를 입력하는 예인데 T_{target} 이 <> 구조 정보를 가지기 때문에 [표 4]의 규칙에 따라서 (b)의 ㉠ 또는 ㉡와 같은 결과를 예측할 수 있다. 교정자는 마우스 휠을 회전하여 두 결과를 화면상으로 보며 원하는 결과를 선택한다.



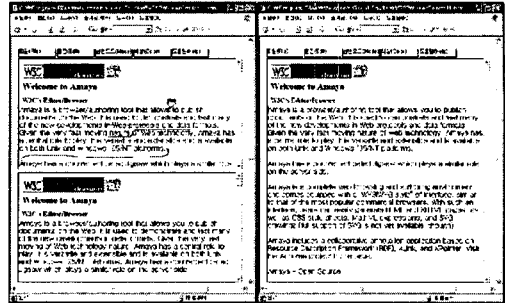
[그림 2] Transpose 명령에 의한 구조정보 변경

5. 교정 시스템 구현

본 연구에서는 제안된 규칙 기반 기법을 기반으로 하는 교정 시스템을 구현하였으며 교정 대상 문서는 일반 HTML 문서를 사용하였다. 사용의 편이성과 향후 다중 사용자간의 협업 작업을 고려하여 Internet Explorer 에 ActiveX 형식으로 설치하여 사용한다. 본 장에서는 이러한 구현 결과 중 제안 기법의 적용 결과를 설명하도록 하며 시스템의 전체 화면은 다음 [그림 3]와 같다.

제안 시스템에서는 교정 부호 입력을 위하여 펜 기반의 입력 인터페이스를 제공한다. 교정 시스템은 교정 부호 입력 인터

페이스 외에 연산이 실행된 결과물도 바로 확인할 수 있어야 하므로 [그림 2]의 (a)와 같이 입력된 부호의 실행 화면을 같이 보거나 (b)와 같이 입력 즉시 실행되는 화면을 제공하고 있다.



(a) 실행 결과창 분리 모드 (b) 즉시 실행 모드
 [그림 3] 교정 시스템 인터페이스

6. 결과

본 연구에서는 전통적 교정 모델을 기반으로 하는 펜입력 교정 환경에서 보다 정확한 텍스트 편집 기법을 제시하였다. 이를 위하여 먼저 펜기반 교정 시스템 모델링을 통하여 교정 부호와 편집 대상 텍스트의 속성을 정의하고 분류하였다. 정의된 모델을 이용하여 펜기반의 교정 부호와 편집 대상 텍스트간의 영역 매칭과정에서 발생하는 모호성을 해결하기 위하여 텍스트의 영역을 다양한 기준선으로 세분화 하고 입력된 교정 부호의 특성 및 입력 영역에 따라 서로 다른 규칙을 적용하여 텍스트를 글자, 단어, 문장, 절 등의 영역으로 인식하도록 하였다. 결과적으로 모호한 교정 부호의 입력을 인지적인 관점에서 해석하여 전통적 교정 모델의 교정 환경에서 보다 정확한 교정 작업 수행을 지원하도록 하였다.

또한 현재 대부분의 교정 작업은 HTML 이나 XML 과 같은 웹 기반의 구조문서 환경에서 교정이 이루어지므로 제안된 모델을 이용하여 구조 정보 변경 규칙을 적용할 수 있다. 향후 본 연구진은 문서 구조 변경을 위한 규칙 모델 정의와 다중 사용자 환경에서의 협업작업 및 문서 변경 탐지를 통한 문서 버전관리와 관련하여 연구를 수행할 예정이다.

참고문헌

- [1] David K. Farkas and E. Plotrock, Online editing, markup models, and the workplace lives of Editors and Writers, IEEE Trans on Prof. Communication (1995)
- [2] Gary Hardock, Gordon Kurtenbach, A marking based interface for collaborative writing, ACM UIST, (1993)
- [3] Jacques Andre, Helene Richy, Paperless editing and proofreading of electronic documents, Proc of EuroTex, (1999)
- [4] Amaya Homepage, <http://www.w3.org/Amaya/>, (2002)
- [5] Janis Ramey and Marlene Miller, High-Tech Publications Need Old-Fashioned Editing, IEE Technology Teamwork, (2000)
- [6] Dean Rubin, Specifying gestures by examples, ACM Computer Graphics Volume 25. p329-337, (1991)