

카메라를 이용한 flowchart - C변환 시스템

Flowchart - C Conversion System using Camera

이 창 우, 주 윤 희, 손 영 선
동명정보대학교 정보통신공학과

Chang-Woo Lee, Yun-Hee Ju and Young-Sun Sohn
Department of Information & Communication Engineering,
Tongmyong University of Information Technology
E-mail : yssohn@tmic.tit.ac.kr

ABSTRACT

본 논문에서는 CCD 흑백 카메라를 이용하여 프로그래머의 알고리즘이 표현된 flowchart의 영상을 입력받아 C언어 코드로 변환하는 시스템을 구현하였다. 입력된 영상을 이진화 처리한 영상으로부터 flowchart 기호들을 인식하기 위하여 chain code 방법을 이용하였고, flowchart 기호에 기술된 영문자 및 특수문자의 인식을 위하여 가로 및 세로 히스토그램을 이용하여 한 문자씩 분할한 후 각 문자들을 구성하는 흑화소 pixel의 합과 chain code 방법을 사용하였다. 가로 및 세로 투영을 이용하여 흐름선을 인식 함으로써 flowchart의 논리흐름을 파악할 수 있었다. 이 시스템을 수치연산에 적용하여, 프로그래머의 알고리즘에 부합하는 프로그램이 작성 되어짐을 확인할 수 있었다.

키워드 : C언어, flowchart, chain code, 문자인식

1. 서론

소프트웨어 개발 목적에 따른 많은 프로그래밍 언어 가운데 C언어는 자연어에 가까운 고급 언어이며, 순차적 처리에 뛰어난 장점을 갖고 있다[1,2]. 그러나 C언어를 이용하여 사용자의 목적에 맞게 프로그램을 작성하기 위해서는 문법구조 및 번역 소프트웨어의 사용법을 숙지하고 있어야 하며, 자신의 알고리즘을 프로그램에 맞게 기술해야 하는 어려움이 있다[3]. 그래서 프로그래머의 알고리즘으로 프로그램 코드를 생성시키는 방법에 대하여 많은 연구가 행하여지고 있다[4,5]. 그러나 이 연구들은 사용자의 알고리즘을 직접적으로 C 코드로 변환 시켜주는 것이 아니다. 최근에는 영상 입력을 통하여 문서를 자동으로 입력시켜 처리해주는 시스템이 연구되어지고 있다[6,7,8]. 본 논문에서는 프로

그래머의 알고리즘이 표현된 flowchart를 CCD 카메라를 이용하여 영상으로 입력 받아 flowchart 기호, 문자 및 흐름선을 인식하여 C언어로 변환, 출력 및 저장하는 시스템을 구현하였다.

2. 시스템 알고리즘

본 논문의 시스템 알고리즘은 그림 1의 시스템 흐름도에서 알 수 있듯이 흑백 CCD 카메라를 이용하여 프로그래머의 알고리즘이 표현된 flowchart 영상을 입력 받아 이진화 하는 전처리 과정을 거친다. 이진화 처리된 영상에서 경계선을 추출한 후 chain code 방법을 이용하여 flowchart 기호를 인식하였다. 문자를 인식하기 위하여 flowchart 기호를 제거하고, 가로 및 세로 히스토그램법을 이용하여 한 문자씩 분할하였다. 인식속도 향상을 위해 문자의 세로 길이에 따라 크게 대/소문자를 분류한 후, 흑화소 pixel의 합과 chain code 방법을 사용하여

문자 인식을 하였다. 흐름선을 인식함으로써 flowchart의 논리적인 흐름을 파악하였으며, 입력된 flowchart 영상의 기호, 문자 및 흐름선 인식이 끝나면 파악된 데이터를 통해 C 코드로 변환시켜 출력 및 저장하였다.

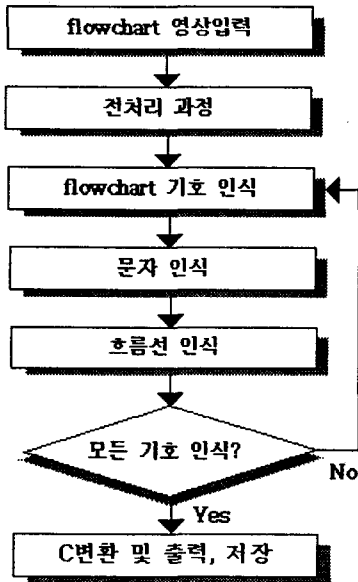


그림 1. 시스템 흐름도

3. flowchart 기호 인식

입력된 영상의 이진화 처리와 경계선 추출 및 세션화한 후 외곽선을 제거하여 flowchart 기호와 문자만 남은 영상을 얻을 수 있다. 가로 투영을 이용하여 각 flowchart 기호의 시작점을 추출하고 chain code 알고리즘을 이용하여 flowchart 기호를 인식하였다.

3. 1 전처리 및 경계선 추출법

CCD 카메라로 입력받은 영상의 인식률을 높이기 위하여 수회의 실험을 통해 얻은 임계값을 적용하여 이진화 처리하였다. 경계선 추출법은 영역의 경계를 나타내는 특징으로 픽셀들의 밝기가 갑작스럽게 변하는 부분을 추출하는 방법이다. 경계선 추출을 위하여 연산속도가 빠르고 사용하기 쉬운 3×3의 라플라시안 마스크를 이용하여 그림 2와 같이 경계선을 추출하였다 [9]. 경계선이 추출 되면 세션화를 거친 후 외곽선을 제거하여 기호와 문자만 남은 영상을 얻을 수 있다.

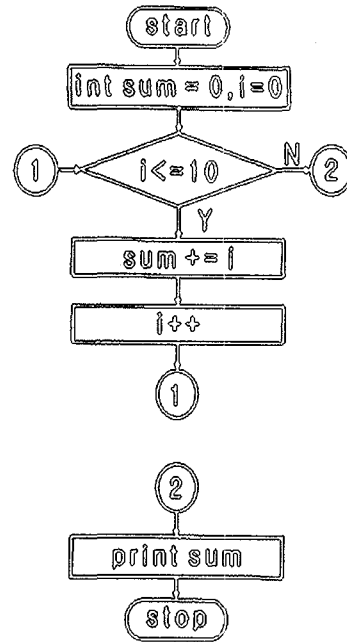


그림 2. 경계선이 추출된 영상

3. 2 flowchart 기호의 시작점 추출

외곽선이 제거된 영상에서 flowchart 기호를 인식하기 위하여 그림 3과 같이 가로 투영을 이용한다. 투영시킨 영상의 기호와 기호 사이에는 흑화소 pixel이 존재하지 않는다는 사실로부터 모든 기호의 시작점을 찾을 수 있다. 한 줄에 기호가 둘 이상 일 때는 세로 투영을 사용하여 기호의 개수를 알아낸다.

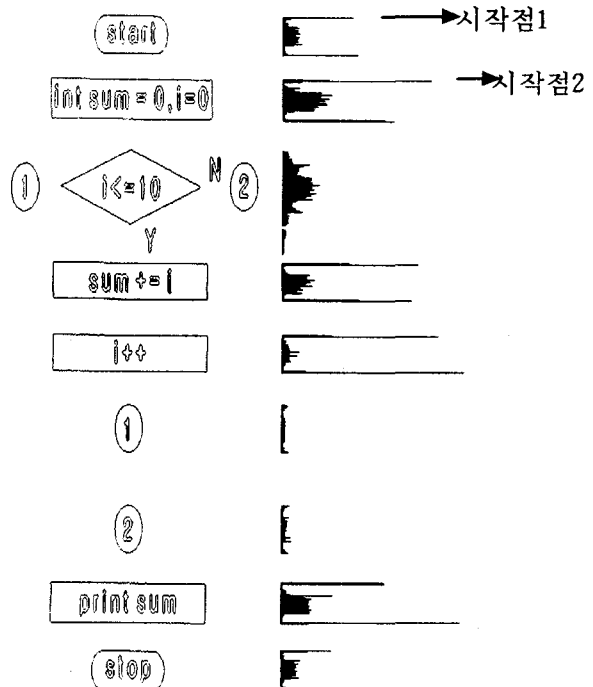


그림 3. 외곽선 제거 영상 및 가로 투영한 영상

3. 3 chain code

chain code는 그림 4에서 보는 바와 같이 기준 픽셀에 대한 인접 픽셀의 방향을 코드로 나타내고, 이 코드를 이용하여 인접한 연결 픽셀들을 4가지 방향 또는 8가지 방향으로 표시한다. 연결된 픽셀들의 시작점만을 그 행과 열 번호를 이용하여 위치를 표시하고 나머지는 그 픽셀에 대한 방향을 코드로 나타냄으로써 주어진 윤곽선을 단일하게 나타낼 수 있다[10]. 본 논문에서는 도형 특성에 대한 세밀한 정보를 얻기 위해 8-이웃 chain code를 이용하였다.

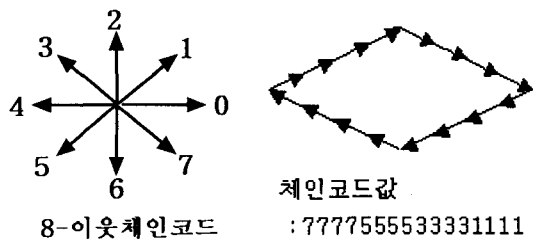


그림 4. 8-이웃 chain code

4. 문자 인식

그림3의 외곽선이 제거된 영상에서 기호 인식 처리 후, 기호를 이루는 픽셀을 제거하면 문자만 남게 되고 가로 투영을 통하여 문자의 영역을 검출하고, 세로 투영을 이용하여 각 문자를 분할한다. 세로 투영된 영상에서 각 문자를 이루는 흑화소 pixel의 합과, 투영하지 않고 경계선만을 추출한 문자의 chain code값을 표준 문자 패턴과 비교하여 문자를 인식하였다.

4. 1 영역 검출 및 문자 분할

영역 검출은 그림 5에서 보여지듯이 가로 투영을 통해 문자열이 존재하는 영역의 좌표점을 찾아내었고, 문자 분할은 세로 투영을 이용하였다[8].



그림 5. 가로 및 세로 투영 영상

4. 2 문자 인식

문자 분할을 통해 떼어진 영상을 그림 6과 같이 각 문자 영역의 최고점에서 아래로 그리고

최하점에서 위로 이동하면서, 각각 첫 번째의 흑화소 위치 좌표를 검출하여 그 차로 구한 문자의 세로 길이를 이용하여 크게 대/소문자, 특수문자로 분류하여 인식의 속도를 높였고, 흑화소 pixel의 합과 chain code값을 표준 문자 패턴과 비교하여 문자를 인식하였다. 그러나 [(영문), 1(숫자)], [p, b], [a, e, s]등과 같이 인식이 애매한 문자들은 정확한 문자 인식을 위하여 가로 또는 세로 투영한 영상의 분포를 이용하였다[8].

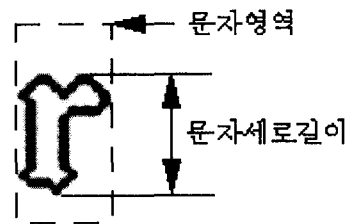


그림 6. 경계선 추출한 문자 영상

5. 흐름선 인식

세로 방향의 흐름선은 하나의 flowchart 기호의 종료점과 다음 기호의 시작점을 이용하여 흐름선 영역을 찾고, 영역에 대해 가로 투영한 영상의 중점을 찾아 두 부분으로 나누어 흑화소 분포가 큰 쪽을 흐름선의 종료점으로 인식한다. 가로 방향의 흐름선은 세로투영을 적용하여 흑화소가 존재하면 흐름선이 존재하는 것으로 인식하고 그렇지 않으면 흐름선이 존재하지 않는 것으로 인식한다. 흐름선의 존재가 인식되어지면 세로 방향 흐름선 인식과 같은 방법을 사용하여 흐름선의 방향을 인식하였다.

6. C 변환 및 출력

입력된 영상의 모든 flowchart 기호, 문자, 흐름선을 인식한 데이터를 연결형 리스트의 각 노드에 저장한다. 모든 노드의 인식문자를 검색하여 각종 함수에 따라 해당 헤더파일을 추가하고, printf()나 scanf()등의 입출력 함수에 관한 변환을 거친다. flowchart 기호에 의한 변환은, 예를 들어 기호명이 선택기호이고 흐름선이 세 개 존재한다면 흐름선이 나가는 쪽의 연결기호 번호와 아래의 연결기호 번호의 일치함을 판단하여 while문으로 변환하는 것과 같이, 모든 기호에 대해 흐름선의 개수와 기호의 종류에 따라 그림 7의 변화과정과 표1의 변환방법에 의해 각각 C변환 하였다. 모든 기호에 대한 변환이 끝나면 들여쓰기 설정을 거쳐 결과를 출력 및

저장하였다. 예로써, 그림 2에서 보여지는 '1에서 10까지의 합'에 관한 flowchart를 본 시스템에 적용한 결과가 그림 8에 보여진다.

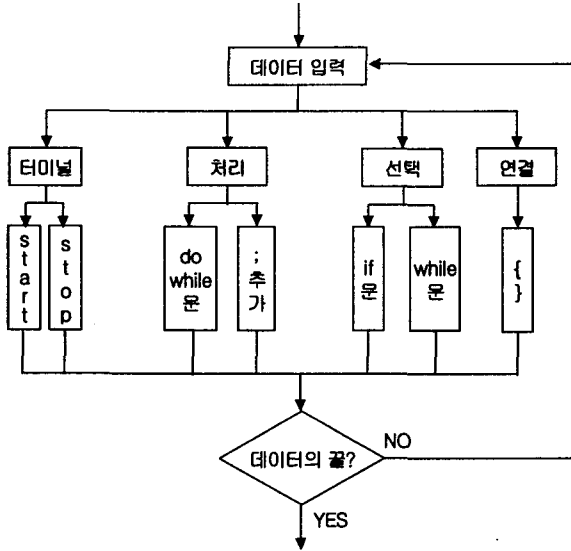


그림 7. 기호에 따른 변환 블록도

표1. flowchart 기호에 따른 C언어 변환

flowchart기호	기호명	변환 방법
	터미널	start, stop를 구분하여 int main() 또는 return 0; 를 삽입한다.
	처리	입력되는 흐름선이 있다면 do~while로 처리하고 나머지는 :만 붙인다
	선택	기호로부터 나가는 흐름선이 두개이면 if문(조건문), 세개이면 while문(반복문)으로 변환
	연결 기호	위, 아래의 다른 기호의 연결 번호와 비교하여 반복문이나 조건문 검증, {}를 삽입한다.

그림8. C변환 코드 출력

7. 결론 및 향후 과제

본 논문에서는 CCD 카메라를 이용하여 프로그래머의 알고리즘이 표현된 flowchart를 C언어로 변환해 주는 시스템을 구현하였다. flowchart 기호를 인식하기 위해 chain code, 문자 인식에는 세로길이와 흑화소 pixel 합 및 chain code를 이용하였으며, 흐름선을 인식하여 논리적 흐름을 파악하여 C 코드로 변환함으로써 프로그래머의 알고리즘에 부합하는 결과가 얻어짐을 알 수 있었다.

향후 과제로는 Windows환경에서의 인식 시스템의 구현 및 복잡하고 다양한 알고리즘이 표현된 flowchart에 대한 C변환의 검증이 고려되어진다.

참고문헌

- [1]권호열, "소프트웨어 개발 프로세스의 연구동향", 한국 정보 과학회지, Vol. 20, No. 1, PP. 0006~0014, 2000
- [2]원유현, "프로그래밍 언어론", 정익사, 2001
- [3]최근수, 하상호, "C해석기를 통한 C 프로그램 실행의 시각화", 정보처리학회, Vol. 9, No. 2, PP. 297~300, 2002
- [4]곽상훈, 이동익, 배영환 "실시간 운영체제 환경을 고려한 SDL명세로부터의 C코드 생성", 정보과학회, Vol. 28, No. 02, PP. 469~471, 2001
- [5]선은두, 김희천, 우치수, "구조적 설계를 통한 C코드 생성", 정보과학회, 1994
- [6]오일석, 김수형, "문서 영상 처리 기술과 정보 검색", 저널 IPIU2002, Vol. 14, No. 01, PP. 0411~0416, 2002
- [7]신일식, 정향영, 손영선, "원형 패턴 벡터 알고리즘을 이용한 손가락 이동에 의한 커서제어", 퍼지 및 지능시스템 학회, Vol. 11, No. 6, PP. 487~490, 2001
- [8]곽주원, 강순대, 백승복, 박상욱, 박정현, 박홍선, 손영선, "인간에게 친밀한 인쇄체 한글 영상의 문서화 시스템", 퍼지 및 지능시스템 학회, Vol. 12, No. 2, PP. 361~364, 2002
- [9]장동혁, "Visual C++을 이용한 디지털 영상처리의 구현", 정보게이트, 2001
- [10]박종현, 박순영, 조완현 "효율적인 내용기반 영상검색을 위한 개선된 chain code 와 HMM", 정보산업연구지, Vol. 8, No. 1, PP. 23~33, 2000