

인쇄된 악보의 음표인식에 관한 연구

이 창현*, 권 호열*, 이 상희*, 김 백섭**
*강원대학교 전자공학과, **한림대학교 전자계산학과

A Study on the Printed Music Note Recognition

©C. H. Lee*, H. Y. Kwon*, S. H. Lee*, B. S. Kim**
* Dept. of Electronics Engineering, Kangwon National Univ.
** Dept. of Computer Science, Hallym Univ.

ABSTRACT

In this paper, we proposed an algorithm for the musical note recognition. Firstly, a given bit-mapped music score image is converted to a set of individual note pattern images via vertical projection. Then, the pitch of a note is determinal by comparison in the note-head position with the reference five-lines. Also, the length of a note is found via leader clustering with a set of normalized note patterns. Finally, a datafile to play the music is obtained using the pitch and length of musical notes.

Experimental results with a simple musical score image show that the proposed scheme is performed well.

I. 서론

최근의 멀티미디어 기술은 종래의 수치계산만을 전문으로 하던 컴퓨터가 문서, 그래픽, 영상정보와 더불어, 음성/음악 정보까지 포함하는 복합매체 정보도 효율적으로 처리 할 수 있는 가능성을 보여주고 있다. 이 가운데 컴퓨터를 이용한 음악정보 처리기술은 다양한 음악관련 유틸리티의 제공을 통하여 전문 음악인의 활동 영역이었던 작곡, 편곡, 및 연주분야를 일반인들도 쉽게 참여하여 취미생활로 즐길 수 있도록 바꾸어 놓았을 뿐 아니라, 음악 정보를 디지털 형식으로 저장 및 재생하는 기능을 이용하여 음악전용 데이터베이스, 또는 음악과 영상의 복합매체 데이터베이스의 구축으로 발전하고 있다. 이와 관련하여 클래식 명곡을 수록한 데이터 베이스가 이미 유통되고 있다.

위와 같은 컴퓨터음악 정보의 저장형식은 음악신호를 음성신호의 경우와 같이 A/D변환을 통하여 얻어진 이진 정보를 저장하는 샘플링 방법과, 악보의 내용을 부호화하여 저장하는 방법으로 대별된다. 샘플링 방법은 정보의 저장과 재생 과정이 간단하다는 장점이 있으나, 저장되어 있는 음악정보의 편집, 변경기능을 구현하기 어렵다. 이에 반해, 악보의 부호화 방법은 기존의 악보를 컴퓨터용 화일로 옮기는 작업상의 난점이 해결된다면 샘플링 방법이 지닌 약점을 해결할 수 있을 뿐만 아니라, 풍부하게 존재하는 악보를 이용하여 음악정보 데이터 베이스를 구축하는데 효과적인 방법이 될 것으로 기대된다. 그러나, 악보를 부호화된 컴퓨터용 화일로 옮기는 입력과정은 대개 오퍼레이터가 악보를 보고 일일이

Coding 해야 하는데, 이 방법은 매우 번거롭고 작업상의 오류발생 문제도 심각하므로, 컴퓨터 음악용 악보화일을 악보로부터 자동생성하는 효율적인 방법의 개발이 필요하다. 이를 위해서 인쇄된 악보로부터 Digitizer(Video Camera/Scanner)를 통하여 입력받은 영상데이터를 연주용의 악보화일로 직접 변환하는 것이 필요하며, 이때 보표, 음표, 박자, 조옮김, 마디 기호 등등의 악보 고유의 패턴들과, 가사부분의 문자패턴을 포함하는 복합적인 패턴인식 알고리즘이 요구된다.

본 논문에서는 인식할 악보가 인쇄된 단순한 형태의 악보라는 가정하에 음악연주에 필요한 최소한의 정보인 음높이와 음길이를 인식할 수 있는 방법에 관해 연구하였다. 먼저, II장에서는 Projection 기법[1]을 사용해 개별 음악부호를 분리하고 이때 얻어진 개별 음악부호에서 음높이의 정보를 얻으며, 한편, 개별 음악부호 영상의 정규화와 Leader clustering[3][4][5][6][8]을 통한 패턴 분류에 의해 음길이를 결정하는 과정등을 고찰해보고 끝으로 III장에서는 컴퓨터 시뮬레이션을 통하여 제안된 알고리즘의 성능을 알아본다.

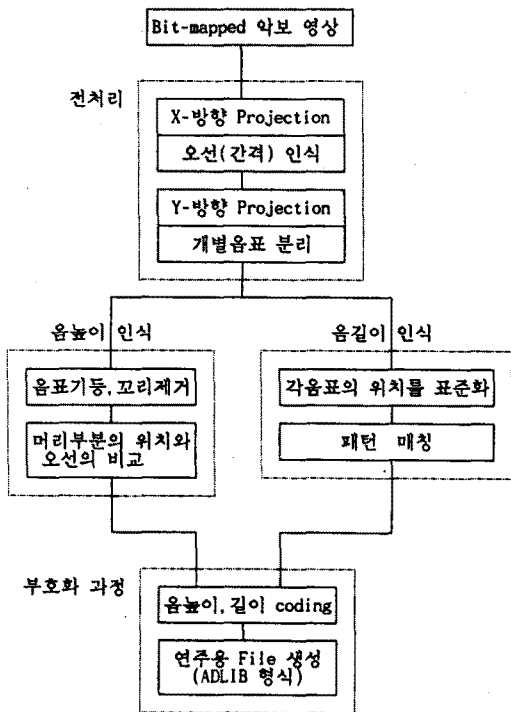
II. 음표 인식 알고리즘

본 연구는 인쇄된 악보로부터 얻어진 <그림-1>과 같은 2진영상을 이용하여 악보내의 음표가 갖는 음높이(pitch)와 음길이(length)를 인식한 후, 컴퓨터 음악 연주를 위한 Data File을 자동적으로 생성하는 것이 목적이다. 이를 위하여, 본 연구에서는 입력데이터로서 단순한 형태의 악보에 대한 2진영상을 사용하고, 음악 연주용 data file 형식은 널리 쓰이는 ADLIB 인터페이스를 기준으로 한다.



<그림-1> 인식할 악보 "학교"

위와 같은 악보의 음표 인식문제를 위한 전체적인 알고리즘의 개략적인 구성은 <그림-2>과 같다.

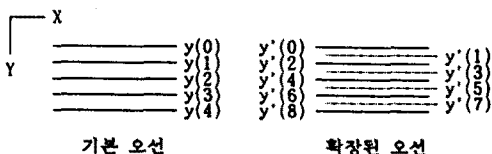


<그림 - 2> 악보인식 알고리즘의 개략적인 블록도

1. 전처리

2진 영상으로 주어진 악보상의 음표가 갖는 음높이와 음 길이에 관한 정보를 인식하기 위한 전처리 과정으로서 음높이의 기준이 되는 오선의 위치를 결정하는 것과 함께, 전체악보영상을 개별 음악부호 패턴영상으로 분할하는 것이 필요하다.

먼저, 음높이의 인식을 위하여 오선을 기준으로 하여 음표머리가 갖는 위치를 결정한다. 음표머리가 오선 위에, 또는 오선의 선간에 위치하는 것을 인식하기 위하여 기본오선을 포함하여 오선의 중간을 지나는 4개의 선을 포함하는 9줄의 확장된 오선으로 변환한후<그림-3>, 음표의 머리의 중심이 갖는 Y좌표가 확장된 오선 가운데 어느선에 가장 가까운지를 조사하여 그중에 해당하는 음의 높이를 읽는다.



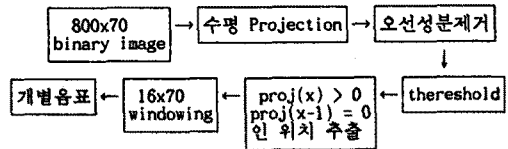
<그림-3> 오선의 간격을 분할

수평방향으로의 Projection을 통하여, 음높이의 기준이 되는 기본오선의 좌표 $y(m)$, $m = 0, 1, \dots, 4$, 를 추출한다. 이때의 추출된 오선정보로부터 최상단의 직선과 최하단의 직선을 인식하여 줄간의 간격을 결정하고, 식(1)에 의해 확장된 오선의 좌표 $y'(m)$, $m = 0, 1, \dots, 8$, 를 결정 한다.

$$y'(k) = y(0) + k*[y(4) - y(0)]/8, \quad k = 0, 1, \dots, 8$$

----- 식 (1)

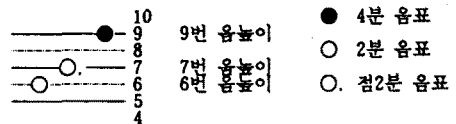
악보영상에 대해 수직 방향의 Projection을 통하여 투영 데이터를 얻고, 개별음표의 투영데이터가 모여있는 특징을 이용하여 개별음표를 분리한다. 이때, 오선성분은 모든 개별음표에 대해 규칙적인 투영데이터로 나타나므로 Threshold 값을 취해 제거 한다. 이와 같이 800x70 악보영상을 16x70 pixel의 크기를 갖는 개별음표 영상으로 분할한다.



<그림-4> 개별 음표분리

2. 음높이의 인식

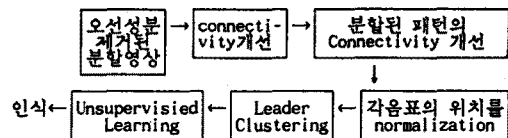
분리된 개별음표에 대해 수평방향으로 Projection을하고 음표의 기둥 및 꼬리부분을 적절한 threshold를 취해 제거한 후 음표의 머리 부분만을 추출한다. 투영데이터 상에 5개의 첨두치로 나타나는 보표의 오선성분은 Median Filtering [1][2]으로 제거한다. 이 결과, 얻어진 음표머리의 중심이 갖는 Y-좌표를 구하여 식(1)에서 얻어진 확장된 오선의 좌표와 비교하고, 가장 가까운 위치의 오선을 이 음표가 갖는 음높이로 한다.



<그림-5> 음표머리의 투영데이터를 이용한 음높이 인식

3. 음길이의 인식

음표의 음길이를 결정하기 위해 패턴인식의 문제가 된다. 그러므로, 악보상의 개별음표 패턴들을 위치, 크기, 및 방향에 대해 각각 정규화한 후에 패턴매칭을 한다. 정규화 과정은 오선이 포함된 개별음표 영상을 균일한 크기의 window내에 위치가 정규화된 상태로 고르게 배치되도록 하며, 이때 효율적인 패턴인식을 위해서 오선성분의 제거와 함께, 오선성분과 중복된 음표영상의 손상을 개선시킬 필요가 있다. 한편, 이와 같이 정규화된 음표 bit-map을 이용하여 대표적이고, 서로 다른 특징패턴을 갖는 대표값(Leader)들로 분류하고, 이 leader들과 입력되는 각 음표패턴을 비교 하는 Leader Clustering을 통해 음표를 분류한다.



<그림-6> 음길이의 인식 과정

오선이 제거된 음표패턴은 오선의 영향으로 부분적으로 손상을 입게되며, 이것은 아래와 같은 window 함수를 이용하여 Connectivity를 개선한다.

W ₁	W ₂	W ₃
W ₄	W ₅	W ₆
W ₇	W ₈	W ₉

$$W_5 = \begin{cases} 1 & \text{if } (W_1 \text{ .AND. } (W_8 \text{ .OR. } W_9)) \text{ .OR. } \\ & (W_2 \text{ .AND. } (W_7 \text{ .OR. } W_8 \text{ .OR. } W_9)) \text{ .OR. } \\ & (W_3 \text{ .AND. } (W_7 \text{ .OR. } W_8)) \end{cases} \text{ --- 식 (2)}$$

<그림-7> masking window

크기 및 위치가 정규화된 각각의 입력 패턴들을 분류하기 위해 Hamming Distance[8]인 Sum(h)를 아래식과 같이 계산하여 그 값이 주어진 threshold보다 작으면 같은 cluster로 분류한다. 이때, 각각의 cluster를 대표하는 패턴은 해당 cluster를 처음으로 분류할때의 input 패턴인 leader를 사용하였다.

$$\sum_{y=0}^{69} \sum_{x=0}^{15} |P_A(x,y) - P_B(x,y)| = \text{Sum}(h); \text{ ---- 식 (3)}$$

$$\begin{cases} \text{Sum}(h) < T & : \text{ 같은 cluster} \\ \text{Sum}(h) > T & : \text{ 다른 cluster} \end{cases}$$

P_A, P_B : 분할 영상패턴

4. 부호화

위에서 얻은 음높이 및 음길이에 관한 정보를 이용하여 음악을 연주할 수 있도록 하려면, 미리 정의된 컴퓨터 음악을 위한 장비용 File의 code format에 따라 이들 정보를 부호화 하여야 한다. 본 연구에서는 ADLIB 인터페이스를 출력 형식으로 사용하였다.

III. 실험 결과

본 연구에서 제안된 알고리즘의 평가를 위하여 2진 영상으로 주어진 악보 데이터를 이용하여 음표의 음높이와 음표를 인식하여 컴퓨터 음악 연주를 위한 data file을 생성하는 실험을 수행하였다. 입력된 악보는 <속소리> 카드가 제공하는 악보영상을 grab한후, bit-map file로 변환하여 입력 데이터로 사용하였으며, 이 때의 악보데이터에 대한 실험조건은 다음과 같다.

- 입력받은 악보는 잡음이 없는 인쇄된 악보이다.
- 장조는 기본장조인 "C" 장조(다장조)이다.
- 음표의 구성은 4분 / 2분 / 점 2분 음표로 구성되었다.
- 음높이에 반음올림/내림은 없다.
- 가사는 무시된다.

<그림-1>과 같은 입력 영상에 대하여 실험 결과는 다음과 같다. 수평방향 Projection후에 <그림-9>처럼 오선성분이 균일하게 나타나므로 이때 각각의 오선의 좌표를 얻는다. <그림-10>은 수직방향 Projection후에 median filtering을 통해 오선성분이 제거된후의 히스토그램을 보여주는데 이때, 분할영상을 위한 정보를 얻는다. <그림-11>은 분할영상을 다시 수평방향으로 projection한후 음표의 머리부분과 확장된 오선과<그림-3> 비교하여 얻은 음높이에 관한 hexa code를 보여준다. 한편, <그림-12>는 16x70의 균일한 윈도우 크기로 분할된 악보를 보여주며, <그림-14>는 모든 분할영상에 존재하는 오선성분이 제거된 결과가 나타나는데, 이로 인해

부분적으로 손상된 음표패턴을 식(2)에 따라서 connectivity가 개선된 결과를 <그림-15>에서 보여준다. leader clustering을 위해 각각의 분할영상의 위치가 갖도록 명준화한후<그림-16>, threshold를변화 시키면서 가장 이상적으로 clustering <그림-17>된 대표 패턴에 음길이 code를 부여한후 각각의 분할영상과 매칭시키면 <그림-18>과 같이 길이정보가 부여된 결과가 나타나는데 4는 음표가 4분음표임을 나타내며, 8은 2분음표, 12는 점 2분 음표임을 나타낸다.

< 결과 DATA >

①. X 축 방향 Projection (오선 detect)



<그림-9> X축 방향으로 projection하여 얻은 히스토그램

②. Y 축 방향 Projection



<그림-10> Y축 방향으로 Projection하여 얻은 히스토그램

③ 음표의 머리부분만을 추출하여 음높이를 인식

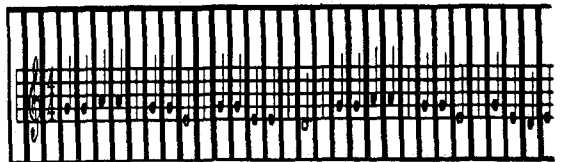


<그림-11> 음높이 인식

④ 영상의 분할

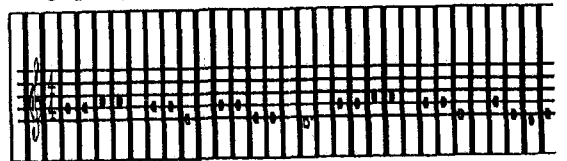
- 위치 Normalization이 않됨

①. 초기 분할



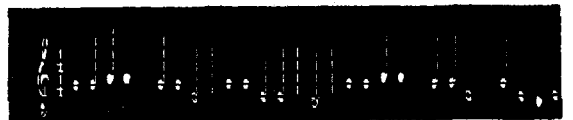
<그림-12> 균일한 window size로 분할된 악보

②. 음표기둥 제거



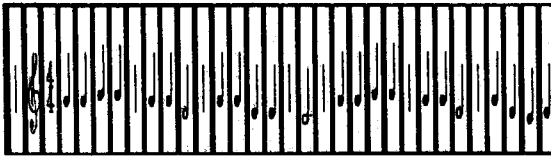
<그림-13> 음표의 기둥부분 제거

③. 오선 제거



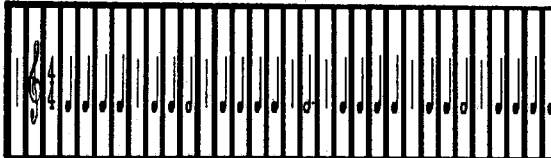
<그림-14> 오선 성분이 제거된 악보

㉔. 오선줄 제거로 인한 connectivity 개선



<그림-15> connectivity 개선

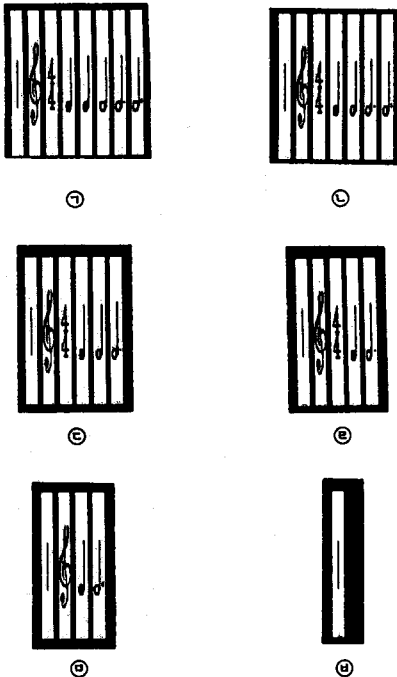
- 위치를 Normalization함



<그림-16> 오선줄 제외한 각 패턴들의 상대적인 위치 평균화

㉕. Leader Clustering 한후 패턴이 분류된 예

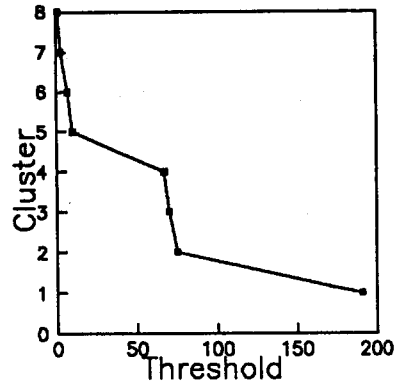
㉖. Threshold의 변화



<그림-17> Threshold에 따라 분류된 Cluster 패턴들

- ㉖ T=1 ㉗ T=3 ㉘ T=7
- ㉙ T=14 ㉚ T=67 ㉛ T=191

㉗. Hamming distance와 Cluster의 변화



㉜. Unsupervised Learning에 따라 박자 인식



<그림-18> Cluster에 따라 Unsupervised Learning후에 음표의 길이 인식

V. 결론

본 연구에서는 비교적 단순한 형태의 악보를 입력 영상 데이터로 하여, 개별음표의 분할을 통한 각 음표의 음높이 및 음길이에 대한 정보를 추출하고, 이 정보를 이용하여 실제 음악 연주에 적용할 수 있음을 보였다.

본 연구에 관련하여 앞으로 연구해야 할 과제는 다음과 같다. 인쇄된 악보를 실제로 Digitizer를 통해 입력하는 경우에 잡음제거와 악보영상의 수평방향 성분이 제대로 설정되지 못한 경우에 대한 전처리 문제가 해결되어야 한다. 또한, 악보영상을 개별 음표영상으로 분할시에 사용하는 Window의 크기가 본 논문에서는 16x70으로 사용하였는데, 영상의 상태에 따라서 적절하게 조정할 수 있는 방법이 필요하다. 이 외에도, 악보영상이 복잡한 음표로 구성되어 있을 경우에 민음표와 민십표의 인식과 잇단 음표, 반음을 포함할 때에 영상분할 문제도 연구되어야 한다.

< 참고문헌 >

- [1] Anil K. Jain, "Fundamentals of Digital Image Processing", Prentice Hall 1989, pp 246-249, 376-377, 362 431-475, 362
- [2] R.C.Gonzalez & Paul Wintz, "Digital Image Processing" 2nd Edition, pp 6-9, 13-134, 162-163, 345-349
- [3] Richard P. Lippman, "An Introduction to Computing with Neural Nets", IEEE ASSP Magazine 1987, pp 4-21
- [4] J.A.Hartigan, "Clustering Algorithm", John Wiley & son New York 1975
- [5] Richard O.Duda & Peter E.Hart, "Pattern Classification and Scene Analysis", Stanford 1973, pp189-190
- [6] John.Hertz, A.Krogh, R.G.Palmer, "Introduction to The Theory of Neural Computation", Addison-Wesley 1991, pp 197-246
- [7] 서 우석, "音樂과 理論", 심설당 1987, pp 7-50
- [8] James A. Freeman, David M. Skapura, "Neural Networks -Algorithms, Applications, and Programming technique-" pp 128-129, 133
- [9] 변정자, "音樂敎育 및 理論과 實踐", 삼호 1988, pp 14-147