Segmentation of Music Primitives

K.C.Ng and R.D.Boyle
Division of Artificial Intelligence, School of Computer Studies,
The University of Leeds,
Leeds LS2 9JT, United Kingdom

Abstract

In this paper, low-level knowledge directed pre-processing and segmentation of music scores are presented. We discuss some of the problems that have been overlooked by existing research but have proved to be major obstacles for robust optical music recognisers [1] to help entering music into a computer, including sub-segmentation of interconnected primitives and identification of nonstraight stave lines, and present solutions to these problems. We conclude that, with knowledge, a significant improvement in low-level segmentations can be achieved.

1 Introduction

Computers are being increasingly used for musical applications, and numerous available musical software package require a machine representation of music to perform their task. Currently, input methods are very time consuming and require some musical knowledge. Optical musical score recognition, especially if able to analyze handwritten scores, would be an interesting and time saving input technique. This is similar to the job of an 'Engraver' who reads a handwritten music score and with specific knowledge transforms it into an engraved music score for printing [5].

This visual problem might seem simple, since writing is black on white paper. Unfortunately, many of the symbols are highly interconnected (Figure 1). Only with musical knowledge can the meaningful figures be discerned.

Figure 1: A number of features which are interconnected.

Interpreting the handwritten notation of a composer is even more difficult because sloppy handwriting results in unclosed and ambiguous note heads, stems not attached to note heads, beams looking similar to slurs, with some phrase marks tying a number of stems and joining up all these features.

The overall target process is thus as follow :

BMVC 1992 doi:10.5244/C.6.49

- A score is scanned optically, and the digitised image fed into the computer as a raster image.
- The computer performs low-level processing; thresholding and deskewing.
- Segmentation; locate and erase the staves and decompose any composite features until they are recognisable as a primitive.
- Recognition; classified a primitive feature, and
- output it in some appropriate representation, of which standard MIDI file [6] is perhaps the most popular, being understood by most currently available software, although it is not ideal as an internal representation during processing.

In this paper, we concentrate on the issue of low level primitive segmentation, especially confusion introduced by slurs, ties, phrase marks and beams. This problem has received attention before - in particular, [4] notes that most musical score recognition is amenable to attack by examining vertical and horizontal projection histograms of suitably chosen windows of an image. Our approach is based on this observation, and our results tend to confirm the view that such simply derived observations carry a wealth information (often sufficient) in this domain.

2 Pre-processing

2.1 Thresholding

The continuous-tone image from the scanner is converted into a binary (black and white) image. For this purpose the Iterative Threshold Selection Method of Ridler and Calvard [10] with Lloyd's modification [8] is used.

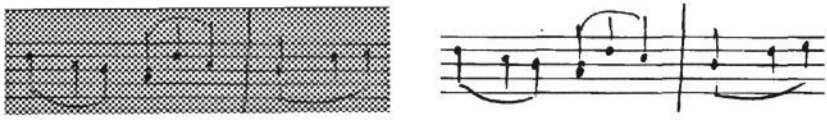

Figure 2: An example input grey tone digital image and the thresholded output.

The threshold method works well and the threshold value converges, usually in less than eight iterations. It produces clean output with little or no noise (Figure 2). Usually, such noise is just some isolated points and can be identified easily.

2.2 Skew correction

At this stage, musical information is contained in the black pixels. In music scores, horizontal alignment is an important property and an indispensable clue during recognition, permitting projection techniques to be used to detect feature position. Symbols are aligned on a five lines (stave lines) staff, and results will only be correct if the staves are horizontal at acquisition time. In

practice there is always a slight skew (characteristically less than two degrees) when the image is captured. We adopt a modification of the approach used by Martin and Bellissant [9] to find the skew angle.

First, we compute a measure of horizontality at some range of possible skew angles with a fixed step, for example, $-5°$ to $+5°$ at $0.1°$ intervals. The middle column of the image, which is the most likely to cut through any horizontal line, is scanned from the top to the bottom row. When we find a foreground (black) pixel, a line template, computed using Bresenham's discrete line algorithm [2], at each possible skew is offered to it, and the number of foreground pixels which fall on the template line counted. The count for each angle is then accumulated for each row, after which the angle with the highest count provides the skew angle. Frequently, there is no clear single peak to determine the skew

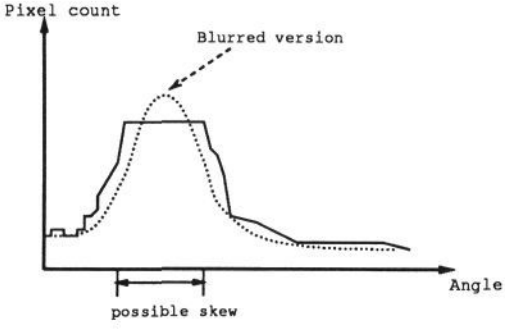

Figure 3: Skew measurement.

(Figure 3). In this eventuality, a Gaussian blur is applied to the responses with sufficiently high σ to make it unimodal - the resulting mode is then accepted as the skew.

Figure 4: An input image with skew, and its deskewed version.

The original 256 grey-level image is then rotated , after which the thresholding algorithm is reapplied. This deskewing process is certainly worthwhile (Figure 4). Deskewed images have more even and smooth stave thickness and lighten the effort of locating and removing them at a later stage. Often, the results are not pixel-perfect, but are at least good enough.

3 Locating and erasing the staves

The stave is the fundamental element of a musical score. A note head by itself may represent the duration of a note, but it has to associate with the stave line to gain its pitch. A staff is a group of five stave lines which are equally spaced, the stave line thickness and the distant between two stave lines being

important parameters at all later stages of recognition. Once we know the position of these lines, they become distractions when we try to recognise the features which have been engraved on or around the staff. Hence, the stave must first be located and measured before erasing it to isolate the musical features. Unfortunately, the stave lines often pass through musical symbols, and so they must be erased selectively in order not to disconnect these symbols.

A histogram of the horizontal projection is generated in which graphs of five equally spaced peaks are usually clear. If they are not (due, for example, to inter-staff text) this pattern can be observed by a suitable blurring of this histogram. From this information, the stave line position, average line width and the space between lines are extracted and recorded.

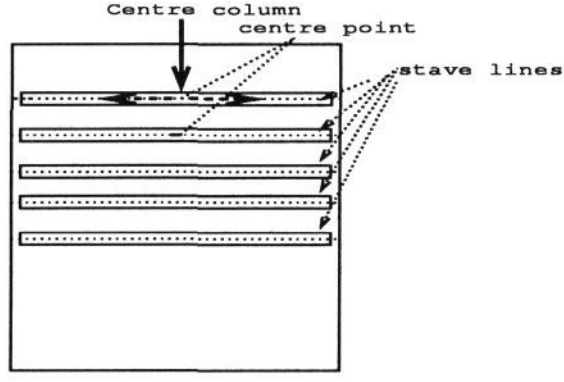

Figure 5: Tracing the stave line.

In practice, the stave lines are not found to be completely straight or of even thickness, causing the parameters for each stave line to differ considerably. Thus we have to repeat the process for each stave line.

For each stave line, we start from its centre column at its central (vertical) position and trace it right and left (Figure 5). Assuming that the line is straight and horizontal, for each column the pixel in the indicated vertical position is taken, and the height of the foreground feature in that column of which it is part is recorded. The distribution of these heights provides a clear mode which provides the 'usual' stave line thickness. Since in practice the line thickness is seen to fluctuate, we choose as the highest acceptable width (W_L), the point after this mode with the distribution gradient almost equal to zero (Figure 6).

Each stave line is then scanned again; commencing from the centre column, the predicted centre pixel is inspected. If it is background, the closest foreground pixel (within the range of $2 * W_L$) to the predicted position in that column is located. If there is no foreground pixel in that range, we assume that the line may be disconnected and go on to scan the next column. If the height of the connected foreground strip so defined does not exceed W_L, its vertical position centre is recorded as the correct best estimate of the line centre position and the strip is deleted. If the strip exceeds W_L in height, it is allowed to remain and the centre estimate not amended. This procedure is then iteratively repeated in neighbouring columns.

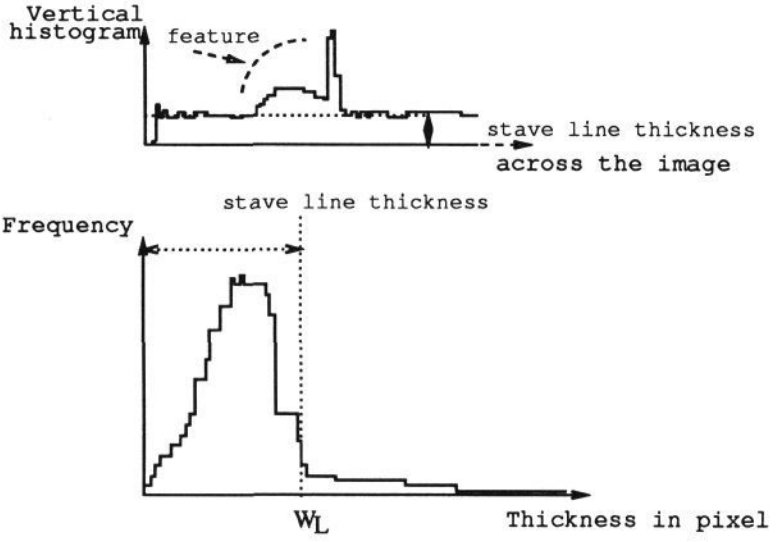

Figure 6: Determining the line thickness.

The output is good but not perfect (Figure 7). Features that were engraved on the staff inherit some noise from the stave lines when they are removed, while thin and long feature such as slurs, ties and phrase marks tend to be disconnected. At this stage, perfect erasing is very unlikely, but these imperfections are not important as they can be overcome during recognition.

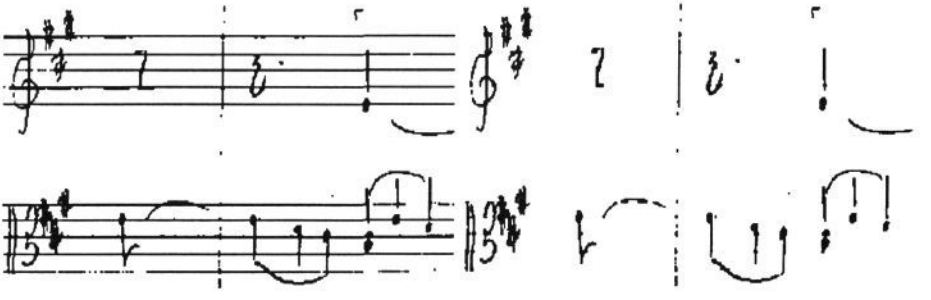

Figure 7: An example of stave line removal.

4 Segmentation

The primitives with which the composer deals (crochets, rests, slurs etc.) are often not as simple for an automatic system to identify as their component parts. For this reason, henceforward we refer to 'primitives' as graphical primitives on the page which sometimes (but not always) do not correspond to the

musical primitives which the destination representation will expect. In particular, stems will be regarded as features to be recognised independently of the note head to which they are connected, and beams connecting quavers (for example) will likewise be regarded as primitives in their own right. Given accurate recognition of these low level primitives, a reliable reconstruction of the derived musical symbols should be straightforward.

When the stave lines are erased, the image will be left with blocks of connected foreground pixels which may be recognisable primitives, such as note heads or stems, or composite objects, such as a group of four semi-quavers, or noise or part of a stave. These are inspected sequentially to determine whether they are primitive features or need further segmentation.

4.1 Primitive sub-segmentation

From the object segmentation, if the object is too 'large' as a primitive feature relative to the staff, it must be a composite made up of a number of connected primitive features (Figure 8).

Figure 8: Examples composite objects.

In practice, the connections are frequently straight lines (beams) or curves (phrase marks, slurs, ties) which cut through the other note stems or connected note heads. Within such composites, a sudden change in vertical projection histogram usually suggests a possible junction point of two separate features.

In a similar application (separation of merged characters during OCR), Kahan et al. [7] observe that maxima in the absolute value of the second difference of the projection is a good indicator of these positions, and that, since 'break points' may be expected to be thin, the ratio of this difference to the projection height is a better measure still. Consequently, we evaluate the measure $(V(x-1)-2V(x)+V(x+1))/V(x)$ across the vertical projection $V(x)$, (Figure 9).

When the horizontal position (X) with the maximum value of this function is found, we assume that this is a junction point at which the object may be separated into two or more smaller features which are connected by a possibly long and relatively thin feature. Instead of just separating the object into two, we attempt to trace and extract the connecting feature. Starting from X, we trace to its left and right until the image boundary is met, or there is no foreground connected pixel ahead. First, find the centre position (interpreted as above) of the connector at X and get its thickness. For the next column to trace, the first guess of the centre will be that of the preceding one; if this prediction is background, 8-neighbour connectivity is used to find any possible immediately connected pixel - this may occur during a sharp turning point on a curve. If the thickness of the column is less than half of the space between two stave lines, the foreground pixels which connect with the centre pixel are marked as an unambiguous part of a connecting feature; otherwise the column must be shared by two closely neighbouring features and is preserved.

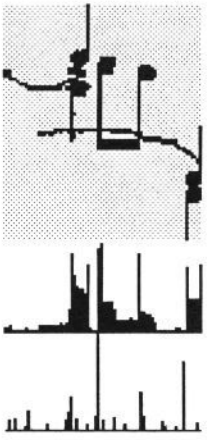

Figure 9: A composite feature, its vertical projection and the ratio of second difference to the projection.

The features we are tracing are either linear (in the case of beams) or approximately quadratic (in the case of slurs, phrase marks and ties). After a suitable number of columns (characteristically 10) have been processed, we can, via a least square estimate, fit a polynomial $y = a + bx + cx^2$ to the observed centre points which is used to predict the likely feature position in the next column. This permits a sufficiently accurate prediction of feature position 'through' objects such as stems within which accurate measurements cannot be made. Figure 10 shows that the connector, in this case a long slur, was identified and when we separate the slur, other primitives are not disturbed. Figure 11 shows the separation of a phrase mark joined with a note head and an accent sign. Notice that some of the estimated centre points of the segmented phrase

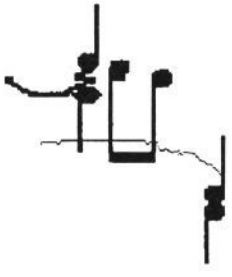

Figure 10: Connector was identified (thin line).

mark are not continuous with their neighbours; this is due to the least squares estimate being insufficiently accurate and falling into background. When this happens, we try to reuse the previous centre position. It is possible that this problem would be solved by a higher order curve approximation.

This process is repeated until the output sub-segment is a possible primitive feature. The termination criteria is the feature having density within its

Figure 11: The sub-segmentation routine, segment out the connector.

bounding box higher than 75%, or being recognisable as a basic primitive such as a note head.

This works very well for phrase marks, slurs, ties and beams if a good break point can be identified. In practice, noise by the side of a stem or bar line may be indicated as the break point; this happens rarely, and is identifiable from its size (very small relative to stave line thickness) and we may simply continue the process.

For vertically connected features, such as a chord, we try to apply the same technique to the horizontal histogram, but the response is not so clear. By making use of the knowledge of the inter-stave line distance and the estimated break points we can deduce good estimates of the location of note heads; possible break points which fall on a stave line or halfway between two stave lines are likely candidates (Figure 12). A complete and robust segmentation of such connected features is likely to require fuller interaction with the recognition phase, and suitable feedback, and this represents work in hand.

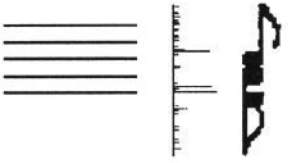

Figure 12: Example input with some estimated break points and its original staff position. The three strong peaks are in 'good' positions.

5 Conclusion

In this paper, we have discussed some potential problems encountered in the early processing of musical scores, and proposed some solutions to them. We have chosen to interpret symbols at the most primitive of levels, and have attempted to segment out such primitives from features which are often highly interconnected, and have demonstrated success at extracting long connecting features such as phrase marks, ties, slurs and beams, leaving the primitives isolated for a subsequent recogniser to work on. Features closely connected in a vertical direction respond to a similar approach, exploiting knowledge of the score geometry. When this approach is combined with a higher-level process providing recognition, we expect to be able to take advantage of the musical syntax [3] to verify the identity of features, or to provide feedback to questionable segmentation, thereby making the whole system very robust and reliable.

References

[1] M Brown A Clarke and M Thome. Problems to be faced by developers of computer based automatic music recognisers. In *International Computer Music Conference*, pages 345–347, 1990.

[2] J E Bresenham. Algorithm for computer control of a digital plotter. *IBM Systems Journal*, 4(1):25–30, 1965.

[3] H Fahmy and D Blostein. A graph grammar for high-level recognition of music notation. In *First International Conference Document Analysis and Recognition*, pages 70–78, September 1991. Sep 30 - Oct 2.

[4] I Fujinaga. Optical music recognition using projections. Master's thesis, Ma McGill University, 1988.

[5] W Gamble. *Music Engraving and Printing.* Da Capo Press Music Reprint Series. Da Capo Press, 1923.

[6] International MIDI Association. *Standard Musical Instrument Digital Interface Files 1.0*, July 1988.

[7] S Kahan, T Pavlidis, and H S Baird. On the recognition of printed characters of any font and size. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-9(2):274–288, March 1987.

[8] D E Lloyd. Automatic target classification using moment invariants of image shapes. Technical Report RAE IDN AW126, Farnborough, U.K., December 1985.

[9] P Martin and C Bellissant. Low-level analysis of music drawing images. In *First International Conference Document Analysis and Recognition*, pages 417–425, September 1991. Sep 30 - Oct 2.

[10] T W Ridler and S Calvard. Picture thresholding using an iterative selection method. *IEEE Transactions SMC*, 8(8):630–632, August 1978.