4. DESCRIPTION OF THE PROGRAM

In this chapter, general workings of the AOMR software is described. The program is divided into seven sections:

1. Staff removal
2. Text removal
3. Segmentation
4. Feature extraction
5. Classification
6. Score reconstruction
7. Learning phase

Given an optically scanned page of a music score, the system first locates and removes the staves. The textual materials, such as lyrics and expression markings are also removed. The remaining symbols on the page are then located and separated from one another for classification. The classification is dependent on the shape of each symbol. The numerical descriptions of the shape are called features, the calculation of which is called the feature extraction. Once the features of the symbol are determined, they are used for classification, which means assigning symbol names to unknown objects. The score is then reconstructed to visually verify the accuracy of the classifier. Finally, the system attempts to improve its performance in the learning phase.

4.1 Staff detection and removal

One of the initial challenges in any OMR systems is the treatment of the staves. For musicians, stafflines are required to facilitate reading the notes. For the machine, however, it becomes an obstacle for making the segmentation of the symbols very difficult. The task of separating background from foreground figures is a unsolved problem in many machine pattern recognition systems in general.

There are two approaches to this problem in OMR systems. One way is to try to remove the stafflines without removing the parts of the music symbols that are superimposed. The other method is to leave the stafflines untouched and devise a method to segment the symbols (Carter 1989, Fujinaga 1988).
In the AOMR system described here, the former approach is taken, that is, the stafflines are carefully removed, without removing too much from the music symbols. This decision was taken basically for three reasons: 1. Symbols such as ties that are very difficult to locate when they are placed right over the stafflines. (See Figure 4.1). 2. One of the hazards of removing stafflines is that parts of music symbols may be removed in the process. But due to printing imperfection or due to damage to the punches that were used for printing (Fujinaga 1988), the music symbols are often already fragmented, without removing the stafflines. In other words, there should be a mechanism to deal with broken symbols whether one removes the stafflines or not. 3. Removing the stafflines simplify many of the consequent steps in the recognition process.

![Figure 4.1 Tie superimposed over staff.](image)

### 4.1.1 The complexity of the process

The following procedure for detecting and removing staves may seem overly complex, but it was found necessary in order to deal with the variety of staff configurations and distortions such as skewing.

The detection of staves is complicated by the variety of staves that are used. The five-line staff is most common today, yet the “four-line staff was widely used from the eleventh to the thirteenth century and the five-line staff did not become standard until mid-seventeenth century, (some keyboard music of the sixteenth and seventeenth centuries employed staves of as many as fifteen lines)” (Gardner 1979, 28). Today, percussion parts may have one to several numbers of lines. The placement and the size of staves may vary on a given page because of an auxiliary staff, which is an alternate or correction in modern editions (Figure 4.2); ornaments staff (Figure 4.3); ossia passages (Figure 4.4), which are technically simplified versions of difficult sections; or more innovative placements of staves (Figure 4.5). In addition, due to various reasons, the stafflines are rarely straight and horizontal, and are not parallel to each other. For example, some staves may be tilted one way and another on the same page or they maybe curved.
Figure 4.2. An example of an auxiliary staff.

Figure 4.3. An example of ornament staves.

Figure 4.4. An example of ossia staff.
Figure 4.5. An example of innovative staff layout.
4.1.2 The reliability of staffline_height and staffspace_height

In order to design a robust staff detector that can process variety of input, one must proceed carefully, not making too many assumptions. There are, fortunately, some reliable factors that can aid in the detection process.

The thickness of stafflines, the staffline_height, on a page is more or less consistent. The space between the stafflines, the staffspace_height, also has small variance within a staff. This is important, for this information can greatly facilitate in the detection and removal of stafflines. Furthermore, there is an image processing technique to reliably estimate these values. The technique is the vertical run-lengths representation of the image.

If a bit-mapped page of music is converted to vertical run-lengths coding, the most common black-runs represents the staffline_height (Figure 4.6) and the most common white-runs represents the staffspace_height (Figure 4.7). Even in music with different staff sizes, there will be prominent peaks at the most frequent staffspaces (Figure 4.8). These estimates are also immune to severe rotation of the image. Figure 4.9 shows the results of white vertical run-lengths of the music used in Figure 4.8 rotated intentionally 15 degrees. It is very useful and crucial, at this very early stage, to have a good approximation of what is on the page. Further processing can be performed based on these values and not be dependent on some predetermined magic numbers. The use of fixed threshold numbers, as found in other OMR systems, makes systems inflexible and difficult to adapt to new and unexpected situations.
Figure 4.6 Estimating staffline_height by vertical black runs. The graph shows that the staffline_height of 4 pixels is most prominent.
Figure 4.7 Estimating staffspace_height by vertical white runs. The graph shows that the staffspace_height of 14 pixels is most prominent.
Figure 4.8 Estimating staffspace_height by vertical white runs with multiple-size staves.
Figure 4.9 Estimating staffspace_height by vertical white runs of a skewed image. The music used in Figure 4.8 is rotated 15 degrees.
4.1.3 The process

The locations of the staves must be determined before they can be removed. The first task is to isolate stafflines from other symbols to find the location of the staves. Any vertical black runs that are more than twice the staffline_height are removed from the original. (See Figure 4.11, Figure 4.10 is the original). A connected component analysis is then performed on the filtered image and any component whose width is less than staffspace_height is removed (Figure 4.12). These steps remove most objects from the page except for slurs, ties, dynamics wedges, stafflines, and other thin and long objects.

The difference between stafflines and other thin objects is the height of the connected component, in other words, the minimal bounding box that contain slurs and dynamics wedges are typically much taller than the minimal bounding box that contain a staffline segment. Removing components that are taller than staffline_height, at this stage, will potentially remove stafflines because if the page is skewed, the bounding boxes of stafflines will also have a height taller than the staffline_height. Therefore, an initial de-skewing of the entire page is attempted. This would hopefully correct any gross skewing of the image. Finer local de-skewing will be performed on each staff later. The de-skewing, here, is a shearing action, that is, the part of the image is shifted up or down by some amount. This is much simpler and a lot less time-consuming than true rotation of the image, but the results seem satisfactory. Here is the algorithm:

1. Take the narrow strip (currently set at 32 pixels-wide) at the center of the page and take a y-projection. Make this the reference y-projection.
2. Take a y-projection of adjacent vertical strip to the right of the center strip. Shift this strip up and down to find out the offset that results in the best match to the reference y-projection. The best match is defined as the largest correlation coefficient, which is calculated by multiplying the two y-projections.
3. Given the best correlated offset, add the two projections together and make this the new reference y-projection. The offset is stored in an array to be used later.
4. If not at the end of the page, go back to Step 2.
5. If the right side of the page is reached, go back to Step 2, but this time move from the center to the left side of the page.
6. Once the offsets for the strips of the entire page are calculated, these offsets are used to shear the entire image. (See Figures 4.13 and 4.14).
Figure 4.10 The original.
Figure 4.11 Vertical black runs more than 2 x staffline_height removed.
Figure 4.12 Connected-components narrower than staffspace_height removed.
Figure 4.13 An example of skewed page.
Figure 4.14 De-skewed image of Figure 4.13 by shearing.
Note that because the run-length coded version of the image is used for shearing, only one operation per column is needed, making the operation extremely efficient.

Assuming now that the image is relatively level, i.e. stafflines are horizontal, taller components, such as slurs and dynamic wedges, are removed. The filter here is still rather conservative, since if a long staff line is still skewed, as a component, it may have a considerable height (Figure 4.15). This precaution is needed because staves on a page are often distorted in different ways.

The result now consists of mostly staffline segments, some flat slurs, and flat beams. At this point y-projection of the entire image is taken again (Figure 4.16). The derivative of the y-projection is used to locate the maxima in the projection (Figure 4.17). Using this information along with the known staffspace_height, the possible candidates for the staves are selected. For each of these candidates, x-projection is taken to determine if there is more than one staff, by searching for any blank area in the projection. Also a rough idea of the left and the right edges of the staff can be determined from the x-projection (See Figures 4.18 and 4.19).

At this point, the run lengths of the region bounding a staff, are calculated in order to obtain a more precise estimate of the staffline_height and staffspace_height of this particular staff. Also, a shearing operation is performed again to make the staff as horizontal as possible.

Using the y-projections employed during the shearing process, the vertical positions of the stafflines can be ascertained. By taking an x-projection of the region defined by the stafflines, the horizontal extents of the staff are determined.
Figure 4.15 Tall connected components removed from Figure 4.12.
Figure 4.16 Y-projection of Figure 4.15.
Figure 4.17 Y-projection (maxima only) of Figure 4.15.
Figure 4.18 An example of staves placed side-by-side.
The next step, knowing the positions of the stafflines, is to remove them. Since the image now consists mainly of staffline segments (Figure 4.20), the strategy is to delete everything but the stafflines; then the image can be XORed with the original image so that, in effect, the stafflines are removed.

At this point, the stafflines are assumed to be flat, so any components taller than the stafflines can be removed (Figure 4.21). This operation differs from the similar operation performed on the entire image, since the more accurate staffline_height that applies to this particular staff is now available.

Also, given the exact positions of the stafflines, components that are between the stafflines are removed (Figure 4.22).

The result is XORed with the original image. Given two bit-mapped images A and A', where A' is a subset of A' (A' is derived from A), an XOR operation has the following important property: All black pixels in A' are removed from A. For example, Figure 4.22 and Figure 4.23 are XORed resulting in Figure 4.24.
Figure 4.22 Objects between the stafflines removed.

Figure 4.23 The original sixth staff of Figure 4.10.

Figure 4.24 The result of XORing Figures 4.22 and 4.23.

Several examples of the staffline removal are shown in Figures 4.25 to 4.36. The time the program takes to remove the stafflines, including reading the input image and writing the resultant image, of 32 pages of different types of music, was approximately 20 minutes, or less than 40 seconds per page on a Sun SPARC 2. All of these image processing, such as filtering and XORing, are performed either on the run-length codes or connected components and not directly on the bit-map, thus making computations extremely efficient.

4.1.4 A note on scanning resolution

The resolution of scanning is 300 dpi (dots-per-inch) which seems to be satisfactory for standard piano music or instrumental parts that may have eight to ten staves per page. The 300 dpi resolution, however, is not fine enough for orchestral scores or miniature scores. For these types of scores, scanning resolution of 600–1000 dpi is needed. Ideally, the thinnest object (usually the stems) should have the thickness of three to five pixels.
Figure 4.25 Stafflines removed from Figure 4.10.
Figure 4.26 Stafflines removed from Figure 4.14.
Figure 4.27 Stafflines removed from Figure 4.18.
Figure 4.30 Stafflines removed from Figure 4.29.
Figure 4.31 The original.
Figure 4.32 Stafflines removed from Figure 4.31.
Figure 4.33 The original.
Figure 4.34 Stafflines removed from Figure 4.33.
Figure 4.35 The original.
Figure 4.36 Stafflines removed from Figure 4.35.