Identifying music documents in a collection of images

David Bainbridge
Department of Computer Science
University of Waikato
davidb@cs.waikato.ac.nz

Tim Bell
Department of Computer Science and Software Engineering
University of Canterbury
tim.bell@canterbury.ac.nz

Abstract

Digital libraries and search engines are now well-equipped to find images of documents based on queries. Many images of music scores are now available, often mixed up with textual documents and images. For example, using the Google "images" search feature, a search for "Beethoven" will return a number of scores and manuscripts as well as pictures of the composer. In this paper we report on an investigation into methods to mechanically determine if a particular document is indeed a score, so that the user can specify that only musical scores should be returned. The goal is to find a minimal set of features that can be used as a quick test that will be applied to large numbers of documents.

A variety of filters were considered, and two promising ones (run-length ratios and Hough transform) were evaluated. We found that a method based around run-lengths in vertical scans (RL) that out-performs a comparable algorithm using the Hough transform (HT). On a test set of 1030 images, RL achieved recall and precision of 97.8% and 88.4% respectively while HT achieved 97.8% and 73.5%. In terms of processor time, RL was more than five times as fast as HT.

Keywords: Optical Music Recognition (OMR), Score Classification, Music Image

1. Introduction

A growing amount of public-domain scanned and photographed music is becoming available on the web. Some of this music is easily accessed through well-indexed library systems, while other music is part of ad-hoc private collections. It is possible to search for these images of music using a search engine, but the results are rather haphazard. For example, Figure 1 shows the results of a Google image search for "Clair de Lune". Three of the first 16 images displayed are scanned sheet music, while the others are clearly irrelevant if one is looking for a music score.

The goal of this research is to provide an automatic filter that distinguishes documents that are likely to be sheet music from other images, which may be photographs or art relating to the title, or more often, album covers and posters relating to recordings and performances of the piece being searched for.

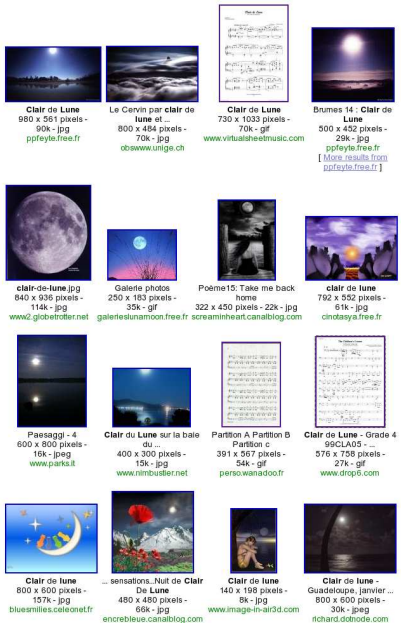

Figure 1. A Google images search for "Clair de Lune", with no filtering.

Such a filter might be used by a search engine (either local or external) to pre-classify documents at the indexing stage, or as a post-filter to eliminate query results just before they are presented to the user. It might also be used as a preliminary phase to OMR so that the recognition is only performed on music pages, avoiding wasted time on title pages and textual commentaries. It could also be used as a desktop search for a personal file space that contains a mixture of music and other images.

Such a system needs to be fast, particularly if being used as a post-filter, to provide good response times. Inevitably speed will be correlated with accuracy, and a fast system may not be completely accurate. Ideally the filter should favour false positives over false negatives (that is, high recall over high precision), since a false positive will just create more work for the user, whereas a false negative will prevent the user from finding a valid music image altogether.

Note that the kind of material to be filtered will depend on how the initial query is formulated. Unfortunately the

user cannot simply add phrases like "music" or "score" to their search, as these may not have been used in conjunction with a music image. Some composer's names are well recognised (e.g. Beethoven) and will generally be a good starting point for a search, although others do not have such distinct names (e.g. "John Williams"), while others share a name with other famous events or people (e.g. the Britten motorcycle). The titles of musical pieces can also be overloaded, such as the "Clair de Lune" example earlier, or any other title that evokes imagery. These issues make a simple filter all the more important for sifting out relevant images.

We have focussed on detecting CMN (common music notation), although have tried to be fairly broad in what we accept as music to allow for applications where the user might be interested in viewing images even if they are low quality or incomplete. Most CMN music is written on five-line staves, and it is these regular lines that are the main feature that we use to detect music. However, the techniques discussed do not require the lines to be in sets of five, so they should detect, for example, 6-line guitar tablature, or 4-line pneume notation.

2. Simple classification methods

There are some straightforward ways to filter music based on simple features of the image file. For example, the size of the image is important; an image that has been captured at less than 100 dpi is unlikely to be very useful, and at this low resolution even just two lines of music on a page would need to be approximately 60,000 pixels, while a full sheet is more likely to be well over 500,000 pixels. This criterion alone will eliminate a lot of photographs and images that are used on web pages.

For example, a Google image search for "Bach fugue" returns about 1,270 images. However, limiting the search to Google's "large" images (the threshold seems to be those with more than about 400,000 pixels) produces just 137 results. While this excludes a number of smaller images that we would recognise as sheet music, the smaller images are generally quite low resolution, mainly used as thumbnails and samples. Of the 137 large images, some 106 are relevant sheet music, which can be expressed as a *precision* of 106/137, or 77%.

Of course, smaller images of music may be of interest too. These can be incipits or sample files that lead the user to sources of the music such as on-line music stores.

Another simple feature might seem to be the use of colour, since sheet music is invariably black and white. However, many historical score collections use full colour images to capture the detail on the page, and even a black and white image might be stored in a format that includes some colour (such as in a frame around the image).

As an example of using colour as a filtering criterion, Table 1 shows the breakdown of the pages returned in the Google search for "Bach fugue" (large images only). Three levels of filtering were used: no filtering ("any colours"), gray-scale only, and bi-level ("black and white") only. The *precision* column shows what percentage of the images retrieved was relevant (i.e. represented images of scores), while the *recall* column shows what proportion of the relevant images were retrieved (assuming that the unfiltered search found all relevant documents). Without any restriction on the colour of the images, 22.6% of the images were false positives, whereas the grey-scale only restriction produced just 2 false positives out of 65 images, and all of the bi-level images were relevant. Unfortunately the latter two filters also produces a lot of false negatives; the grey-scale filter rejected 43 (40.6%) of the relevant images, while the bi-level filter rejected 88 (83%). Moderately good results are achieved in this case by combining the gray and bi-level documents, giving 98% precision and 76% recall, but bear in mind that the user is likely to favour recall over precision.

Filter	Number found	Number relevant	Precision	Recall
None	137	106	77%	100%
Gray scale	65	63	97%	59%
Bi-level	18	18	100%	17%
Gray or bi-level	83	81	98%	76%

Table 1. Precision and recall using for Google image search for "Bach fugue", large images only.

Another problem with using colour as a filter is that for later composers who lived in the period of black and white photography (late 19th and early 20th century), many gray-level images exist of the individual. For example, a search for "Bartok concerto" filtered by gray-level produces only 5 music images out the 47 returned, a precision of just 11% compared with 97% for the "Bach fugue" example. [1]

Thus filtering for colour is more likely to produce false negatives, which is not desirable for this application, although would be useful if the user requires high precision rather than recall.

Another simple approach is simply to run Optical Music Recognition (e.g. [1], [2], [3]) on each candidate page and measure how many valid objects are found, or how many errors occur. However, a full OMR system would take considerably longer to process a page than the simple filters we propose below. The most time-consuming stage is classifying the primitive shapes—note heads, stems, tails, clefs, accidentals and so forth—that constitute a work. This processing cost can be as much as 80 times higher than the time spent locating stave lines [1]. Furthermore, most OMR systems work on bi-level images only, so most of the candidate images would need to be converted before being processed and it is not clear how well these existing algorithms would

[1] It is ironic that composers from earlier periods could be depicted more often using colour images.

work when applied to images that have been converted using standard binary thresholding techniques.

A simpler approach is just to perform a small part of the OMR process, such as stave detection [4], and determine how successful it has been. This is effectively what we have done—generalised to handle grayscale and colour images—with the filters proposed below.

3. Improved classification methods

We have explored a number of properties that can be evaluated quickly and might detect the main features of an image of music, which is primarily the presence of stave lines. Many of these techniques are typical of OMR systems, but in an OMR system one starts with the assumption that the image is music, and hence while they are effective for recognising music, they are not necessarily effective for rejecting images that are not music.

The following methods were not successful in themselves, although may possibly be used in combination with other features for a refined classification.

Horizontal projections : Horizontal projections of a music image usually show peaks where the stave lines are. However, to detect music we need to detect regularly spaced projections, and also this method relies of the image being reasonably straight (the cost of correcting the image would be too high given that we are looking for a simple filter). More importantly, many non-music images (especially text) share the property of having regularly spaced peaks in horizontal projections.

Pixel density : This very simple measure is of some use as music images tend to have approximately 10 to 20% black pixels, compared with text which is around 5%. However, other images can have a very wide range of ratios, and music with wide margins can have an arbitrarily low density, while heavy stavelines and dense writing can give a density up to about 30%.

The following methods have shown promise, and are evaluated more carefully in the experiments below.

Hough transform (HT) : The Hough transform, in its most general form, is a method for evaluating the the likelihood that a parameterised feature exists within an image [5]. Finding straight lines (at arbitrary angles) is a particular instantiation of the transform and one that sees it often used in computer vision work. For our purposes the formulation for straight lines is of particular relevance to our task at hand—locating stave lines.

We use its normalised form for our implementation where a straight line is described using polar coordinates: r and θ describe a line that is perpendicular to the angle θ at distance r from the origin. Values in the $[\theta, r]$ transformed space with high values correspond to strong evidence of there being a line in the original image at that position.

The Hough transform can be computationally expensive, and since in practice we were only interested in horizontal and near horizontal lines the full range of theta was not calculated. Through trial and error we found the range restricted to $\pm 3^\circ$ worked well.

Run-length ratios (RL) : This method involves following vertical scan lines down the page and measuring the ratio of the length of adjacent runs of black and white pixels. As the scan line passes through an empty part of a music stave, the alternating runs of black and white pixels will be the stave lines (black) followed by the gap between the lines (white). Typically the gap (runs of white pixels) is around 5 to 6 times the height of the lines (run of black pixels). Of course, the scan lines will also encounter musical symbols and text on the page, but typically about 25% of the area of the stave lines have no symbols superimposed on them, and so the dominant ratio is that of the height of gaps to the height of the lines. We simply look for the most frequent ratio.

In practice, the ratio for music images can become as small as one to one (for very small score images, which of course are practically unreadable), and as large as 11 to one (where the manuscript lines are very light). In the experiments we allowed for this fairly large range, as even with extreme values it offered good sensitivity for finding music images.

A good feature of this method is that (like the Hough transform) it is not sensitive to small rotations in the image. A disadvantage is that is is not able to detect single-line staves, such as a page of percussion music. Also, it assumes a black and white image, which requires colour and gray-scale images to be thresholded; this is discussed further in the experimental description.

For further discrimination, we included a measure of how common the most frequent ratio was. Typically at least 10%, and more usually 20%, of the samples for music would have the most common ratio, whereas non-music images that had a promising ratio usually didn't have them so dominant. The more sparse the manuscript is, the more dominant the ratio is. Music that makes heavy use of beams seems to have the worst effect on this ratio, but even in this case there will be a number of untouched stave lines.

Hybrid : This method uses run-length ratios (RL) as the initial filter, and then the Hough transform (HT) filter

is applied to the images accepted as music as a secondary filter to eliminate false positives.

An important way to accelerate almost any method that is classifying images is to use samples of each image, rather than working with the entire image. For example, the test could be applied to a number of randomly choses small regions within the page, or for a single, slightly larger region from the centre of the page. This idea was used by Göcke [6], who was performing more detailed classification of music images to identify who the writer was.

For the Hough transform, restricting the angle θ is effectively a form of sampling, but in addition the actual areas of the source image that were transformed was also limited. Various selection strategies were trialled: for example, a random pattern and regions staggered diagonally across the page. Taking pairs of rectangular regions, one located on the left and the other located on the right—a method that echoes that of OMR systems that use horizontal projects to locate stave lines—was found to work reliably across a range of situations. Regions were 120×60 pixels, except where the source image dimension was so small it could not support two side by side. In such cases, the available width was divided in two. Where the source image was tall enough, regions where taken at the top, middle and bottom. This collapsed down to two and then one when space was restricted. For images greater than 500 pixels high, the top middle and bottom approach was extended to two at the top, three in the middle and two at the bottom. The extra lines were also indented by 20%.

For the run-length ratio method, the natural sample to take is a small number of single-pixel wide vertical slices down the page. Because a randomly selected area of a stave is very likely to be unpopulated with notation, the probability of even a single slice providing useful samples is very high, with the main problem being that there is a small chance that a particular slice will encounter barlines or note stems all the way down the page (particularly because it is not unusual for barlines to be aligned even if they are in different systems).

A useful side-effect of performing a constant number of samples for each image is that it avoids spending too much time on large images; some of the larger images contain several megapixels, and it simply isn't worth processing every pixel.

4. Experimental evaluation

The three methods proposed above (HT, RL and hybrid) were evaluated on a set of music that was obtained using the Google image search (all sizes and colours). Five queries were used: "beethoven", "sonata sheet", "sheet music", "mozart", and "overture sheet". Note that inverted commas were not used in the queries. To generate a reasonable cross-section of monochrome, grayscale and colour samples, each query was repeated three times with the appropriate colour-filter on, and the results aggregated. For each query, the images presented on the first five pages were downloaded.

Google presented 1218 thumbnails of relevant images in total for the above queries, although when we downloaded the (source) images, only 1030 were available, that is, 15% of the images had been removed from their web pages since Google indexed them, or else their web server was not responding at the time we ran our sweep. There were only 8 images that were returned by more than one of the five queries made, and these duplicates were removed.

We then manually classified the images to produce a ground-truth database. We needed to be clear on what we mean by images of music. We have focussed on detecting Common Music Notation (CMN) images, excluding other notations such as "piano roll" or textual formats. Also, our goal is simply to detect that the document is likely to be music; we do not attempt to classify it further (for example, Göcke [6] has done work investigating classifying music images to determine who the author is, or one might want to distinguish styles such as tablature or early music).

Even within CMN notation, images encountered include screen-shots of music processing programs, and even a photo of a music stand with music on it! We have used the distinction that we will accept images where the reader can see sufficient CMN music to identify more than three notes or some other substantive musicological information (such as the number of staves); this therefore includes incipits and small samples, as well as original manuscripts and the software screenshots, but excludes a photograph that happens to have some unreadable sheet music as part of the scene.

Of the 1030 available images, the manual classification to establish the ground truth found 367 images that were classified as music, and 663 examples that were not. The images classified as music were those where it was possible in some way to read the notes on the page. For some of the more borderline low-resolution images this would involve a reasonable amount of guesswork to interpret the music, but we tried to be as inclusive as we reasonably could.

An independent set of 336 images were used to fine-tune the filters so that we were not training the parameters on the files that the evaluation was performed on.

4.1. Test sequence

Factoring in the earlier observation about size and use of colour in music (simple classification methods) along with the more specific processing methods that targets the detection of staves, the following test sequence was developed:

1. Is the image big enough?
2. Does it make limited use of colour?
3. Is there a dominant background colour?
4. Test for stave lines (RL, HT or hybrid).

Filter	True Positives	False Positives	True Negatives	False Negatives	Precision	Recall	Total Processing Time
Hough Transform (HT)	357	129	536	8	73.5%	97.8%	205 secs
Run-Length (RL)	357	47	618	8	88.4%	97.8%	36 secs
Hybrid	352	30	635	13	91.0%	96.4%	165 secs

Table 2. Precision and recall and timing results for processing the 1030 test set of images.

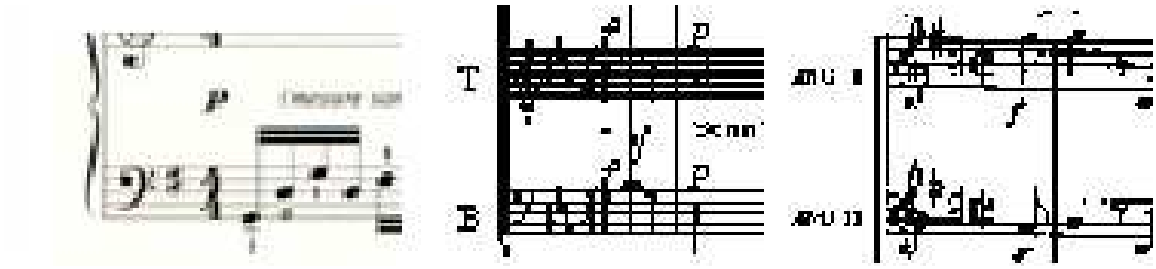

Figure 2. Examples of some of the most borderline images that were classified in the ground truth as readable music; these are typical of the images that caused false negatives.

If an image fails at any of these steps it is immediately rejected. For image size, we set the limit at 120x60 pixels for the smallest image considered since even at low resolutions this is barely large enough to represent more than a few notes. For limited use of colour we used the HSV (Hue Saturation Value) model and calculated a histogram based on the frequency of hue values. Ignoring frequencies counts of less than 10, if the image used more than 100 hues it was rejected since even a colour image of music with illustrations does not exhibit this range of colour. Monochrome and grayscale images trivially meet this requirement as they use only one hue.

For the dominant background colour, the value component of HSV was used. Again a frequency histogram was calculated and if less than 50% of the image values fell within 20 readings of the most frequent value the image was rejected. For sheet music to be readable there has to be a decent contrast between the background (usually white) and foreground pixels. The essence of this test is that, allowing for some discolouration, the majority of the pixels present in the image should be the same. The test was posed in this way in an attempt to handle colour images of vintage sheet music where the paper has often turned a yellowy-brown over time. It also has the advantage that it works equally well for "reverse-video" sheet music (where the background is black and the notation is represented in white), or a blue background for that matter (as occurred in one of the test samples).

If an image passed all these tests, then it was tested for the presence of stave lines using one of the three algorithms under investigation: Hough transform, Run-Length and the hybrid.

Having performed the basic Hough transform, the decision as to whether to classify an image as sheet music or not is based on the number of high values that are found in the transformed space. For each region pair, the maximum value in the transform is retrieved (calculated as a side effect of the basic transform) and it is compared with the combined width of the two regions. If the value is less than 25% of the combined width, then it is rejected outright. If not, then the number of values that are within 70% of the maximum value is computed and this is tested to see if it lies within the range of 1%-25% (exclusive) of the height of the region.

The rationale for these limits is that we are looking for strong evidence of straight lines (within the range $\pm 3°$) that are a modest percentage of the y-dimension. There has to be something there (hence the 1% lower limit) but we don't want it to get too high as this means there are more black lines around than white space.

The RL method was implemented by categorising white to black ratios into a histogram with the ranges 0 to 0.5, 0.5 to 1.5, 1.5 to 2.5 etc., that is, we rounded the ratios to the nearest integer. Most music showed a strong peak at around 5 or 6, but we accepted images where the histogram peak was between 1 and 11 inclusive. A further filter was applied if the ratio was 1 or 2, since these ratios would only reflect a thumbnail image, but might also correspond to effects such as dithering in a non-music image. If the dominant ratio was 2 then if the page was more than 160 pixels high we required the ratio to occur at least 30 times; if it was 1 we simply required the ratio to occur at least 100 times. In addition, the RL method evaluated used just 9 vertical scan samples (see section 4.3).

In line with other steps in the decision-making process, the generation of the HSV histograms is based a coarse sample rather than visiting every pixel in the image. Using a grid formed from 120 pixels evenly spaced out horizontally, and 60 vertically, was found to speed up processing time signif-

icantly, without compromising accuracy.

4.2. Results

Table 2 summarises the results of processing the test set of images with the three algorithms devised, which was run on a 1.7 GHz Pentium processor with 1 GByte of RAM. Both recall and precision are relatively high for all three methods. The hybrid method has more precision, but in the process of improving the precision, it has decreased recall because the extra step removes candidates, but does not add any. Both of the methods involving the Hough transform are significantly slower then the RL method. Given that we are likely to favour recall over precision, the RL method gives excellent performance and takes a fraction of the time to run the tests, and thus is the recommended method.

Many of the false negative results were caused by images that were very low quality, generally with very low resolution and/or faint colours. Figure 2 shows enlargements of samples from images that caused these sorts of problems. The low resolution ones are generally from thumbnails that might be of interest in that they could lead to other images or links that are relevant to the query, even if the music isn't useful in itself. The images with faint colours would probably cause fewer problems if an improved thresholding algorithm were used.

4.3. Sampling for the RL method

An important trade-off for the RL method is the number of vertical scans to make; this affects both the precision/recall and the time taken. Figure 3 shows how these are affected as the number of samples increases. The main concern is the recall, which reaches 97.8% if 9 sample scans are taken; just 8 of the 1030 images were incorrectly rejected. The precision starts dropping beyond this point, despite the increase in samples. This is because the extra samples are likely to pick up other patterns in an image that aren't spread uniformly across the page. The speed is also shown in Figure 3 as a percentage of the test with 25 samples per image. Overall, about 9 samples per image provides an excellent trade-off between the speed, precision and recall. Scanning every vertical line in the images takes about 20 times as long as taking 9 samples per image, doesn't offer any precision/recall benefit, and makes the evaluation time much more dependent on the size of the image.

5. Conclusions

The experiments have shown that it is feasible to provide a fast filter on an image search that distinguishes music images. The run-length ratio (RL) filter is particularly fast, and is able to work with relatively small samples of an image. We favoured false positives over false negatives, and were able to choose parameters for the system that provided good rates for both of these types of errors, giving 97.8% recall and 88.4% precision in our experiments.

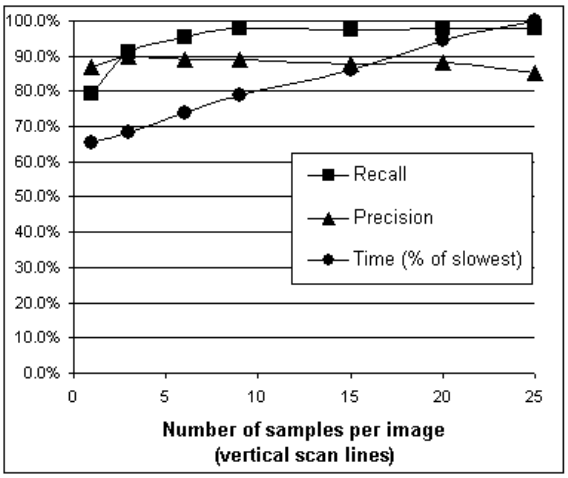

Figure 3. The precision/recall curves and time trade-off for sampling in the RL method.

The Hough transform filter (HT) was almost an order of magnitude slower than the run-length (RL) filter, and didn't show an advantage in the quality of the filtering. Combining both filters into a hybrid did improve the filtering, at the expense of a compromise to the speed of the system. In practice the RL method on its own is likely to be favoured.

Further investigations could include using a machine learning system to evaluate a large number of attributes (such as pixel density, image size, colour range and run-length ratio distribution) to find more accurate ways to classify the images. Better methods for thresholding might also improve speed and accuracy.

References

[1] D. Bainbridge. *Extensible Optical Music Recognition*. Ph.D. thesis, Department of Computer Science, University of Canterbury, NZ, 1997.

[2] K. MacMillan, D. Droettboom, and I. Fujinaga. Gamera: Optical music recognition in a new shell. In *Proceedings of the International Computer Music Conference*, pages 482–485, 2002.

[3] Kia Ng. Optical music recognition for printed music score and handwritten music manuscript. In Susan George, editor, *Visual perception of music notation: on-line and off-line recognition*, pages 108–127. IRM Press, 2005.

[4] Ichiro Fujinaga. Staff detection and removal. In Susan George, editor, *Visual perception of music notation: on-line and off-line recognition*, pages 1–39. IRM Press, 2005.

[5] R. O. Duda and P. E. Hart. Use of the Hough transformation to detect lines and curves in pictures. *Communications of the ACM*, pages 11–15, January 1972.

[6] Roland Göcke. Building as system for writer identification on handwritten music scores. In M.H. Hamza, editor, *Proceedings of the IASTED International Conference on Signal Processing, Pattern Recognition and Applications*, pages 250–255, Rhodes, Greece, 2003. Acta Press, Anaheim, USA.