FAST CAPTURE OF SHEET MUSIC FOR AN AGILE DIGITAL MUSIC LIBRARY

Richard Lobb,
Tim Bell
Department of Computer Science and
Software Engineering
University of Canterbury
Christchurch, NZ
{richard.lobb,tim.bell}@canterbury.ac.nz

David Bainbridge
Department of Computer Science
University of Waikato
Hamilton, NZ
d.bainbridge@cs.waikato.ac.nz

ABSTRACT

A personal digital music library needs to be "agile", that is, it needs to make it easy to capture and index material on the fly. A digital camera is a particularly effective way of achieving this, but there are several issues with the quality of the captured image, including distortions in the shape of the image due to the camera not being aligned properly with the page, non-planarity of the page, lens distortion from close-up shots, and inconsistent lighting across the page. In this paper we explore ways to improve the quality of music images captured by a digital camera or an inexpensive scanner, where the user is not expected to pay a lot of attention to the process. Such pre-processing will significantly aid Music Information Retrieval indexing through Optical Music Recognition, for example. The research presented here is primarily based around using a Fast Fourier Transform (FFT) to determine the orientation of the page. We find that a windowed FFT is effective at correcting rotational errors, and we make significant progress towards removing perspective distortion introduced by the camera not being parallel with the music.

Keywords: image capture, FFT, digital camera

1 INTRODUCTION

Professionally maintained digital sheet music libraries go to considerable trouble to capture clean images that are suitable for a variety of applications (Riley and Fujinaga, 2003). However, increasingly there is a demand for personal digital libraries of sheet music to replace the shelves and briefcases that musicians traditionally use. The personal digital library might be the basis of a digital music stand, and also provides flexible access and backup of material that the musician uses in their day to day work, including the possibility of Optical Music Recognition (OMR) to assist with searching and previewing the music (Bainbridge and Bell, 2001).

While it is feasible to aquire high-quality digital scores from internet stores and web sites, musicians will have a legacy of paper-based music that is not available in digital form (such as personal compositions, transcriptions, and out-of-print works), and these need to be captured (format-shifted) into the digital library.

Such a library needs to be agile, as the librarian is also the user, and often the task of acquisition and cataloguing music will be seen only as an annoying step — witness the state of cataloguing of some musicians' paper based libraries! Furthermore, the format-shifting process may need to occur in a pressured and mobile environment, such as a rough chord chart being written out during a break in a performance due to an unexpected request.

The main way of capturing paper-based music is to use a scanner or a digital camera. A carefully configured scanner can produce excellent reproductions, but a personal scanner is likely to be slow (only one or two pages per minute), not portable, and difficult to use with bulky originals. In contrast a digital camera is very portable and images can be captured in a fraction of a second (Taylor et al., 1999). A four mega-pixel camera gives full-colour capture of a typical page at around 200 dpi, which is quite adequate for reading music from a screen or for optical music recognition. While the resolution of cameras is sufficient for the reproduction of sheet music, unless a carefully controlled environment is used, the quality of the capture is lower because of uneven or poor lighting, and distortions can be caused by the lens and the angle of the camera to the page. Placing the camera directly over a page is generally difficult in an uncontrolled environment because it is likely to create shadows on the page. Distortions in the shape of the document can also be worse than with a scannner; for example, the binding of a book can cause the edge of a page to be curved, and this is not so pronounced on a scanner because the page is compressed onto the scanner glass.

Correcting for these distortions is important for OMR based indexing, for example, because the accurate placement of symbols on the stave is crucial, particularly for identifying the pitch. This may be an issue if a digital camera has been used for capture, or if the image has been stored on microfilm. Even if OMR is not being performed on the music, the quality of the display can be greatly enhanced because even slight distortions in the captured image may produce distracting aliasing effects ("jaggies"), particularly in the horizontal stave lines.

The correction of distortion caused by scanners has received considerable attention in the OMR literature. Distortion correction is generally combined with staff line detection, which is usually performed by either:

- projections, typically horizontal or a series of near-horizontal projections (looking for scan lines with mainly black pixels) e.g. Matsushima et al. (1985); Martin and Bellissant (1991); Baumann and Dengel (1992); Clarke et al. (1988); Fujinaga et al. (1991); Roth (1994), or
- vertical scan lines (looking for regular white gaps between black stave lines) e.g. Carter (1989); Kato and Inokuchi (1990); Reed (1995).

Furthermore, prior work generally assumes that the deformation can be corrected by rotation, partly because music has traditionally been scanned on a flat-bed scanner where rotation is the most likely error.

It is interesting to observe that existing OMR systems generally assume bi-level images, presumably based on the argument that music itself is black and white. However, a gray-scale or colour image contains more information and is therefore more useful for image processing. It also provides a relatively alias-free image for viewing. We argue that for music in a digital library, content should be preserved in colour for as long as possible (storage space permitting) and only converted to bi-level where strictly necessary, for instance when using a bi-level image segmentation algorithm.

In the paper we investigate techniques based on a fast fourier transform (FFT) for removing the distortion that is caused by capturing a page using a digital camera in uncontrolled conditions. We start by considering ways to remove rotation, and then we look at perspective distortion introduced by not having the camera directly above the page.

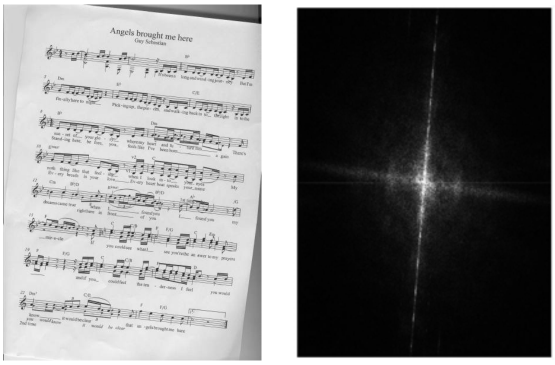
2 FOURIER TRANSFORMS OF MUSIC IMAGES

In this section we show the Fourier transforms of two test images and discuss what information can be obtained from them. We also introduce the idea of windowed Fourier transforms.

2.1 The FFT of an entire rotated page of music

Figure 1a shows a 1700×2340 pixel test image: a page of sheet music scanned with a rotation of approximately 6 degrees as a grey-scale image. Note that the background intensity varies significantly over the image area. Figure 1b shows the magnitude of the Fourier transform of the image in Figure 1a. Throughout the rest of this paper we will loosely refer to the "magnitude of the Fourier Transform" as "the Fourier transform", since we never make use of the phase information of the transform.

In computing Figure 1b the transform has had the zero frequency or "DC component" set to zero and has then been scaled so that the largest remaining component has the arbitrary value of 32. The transform is shifted so that the zero frequency point is in the centre of the image. A white pixel denotes a value of 1 or more, and a black pixel denotes 0. Note the following:



(a) (b)

Figure 1: Test image one, "Angels", rotation only (a) the initially captured image using a flat-bed scanner and (b) its FFT.

- The transform is symmetric about the origin in the sense that $F(-x,-y) = F(x,y)$.[1]
- The most pronounced feature of the image is a line of strong Fourier components at an angle to the vertical axis. The Fourier components on this line arise from the rotated stave lines, so we will call the line the *Fourier Stave Characteristic* or FSC.
- The FSC is made up of the various frequency components that define the set of stave lines. There are two primary frequency components present: a high-frequency component corresponding to the spacing between adjacent stave lines and a low-frequency component due to the spacing between connected lines of music (referred to as systems by musicians).
- The angle between the FSC and the vertical axis can be determined by finding the pixel coordinates of any of the strong components on the FSC well away from the origin. For example there is a strong component at (85,1086) with respect to the origin, i.e., a component with a horizontal frequency of 85 cycles per 1700 pixels (the image width) and a vertical frequency of 1086 cycles per 2340 pixels (the image height). This corresponds to a rotation angle in the original image of

$$\theta = \tan^{-1} \frac{85/1700}{1086/2340} = 6.149^\circ \qquad (1)$$

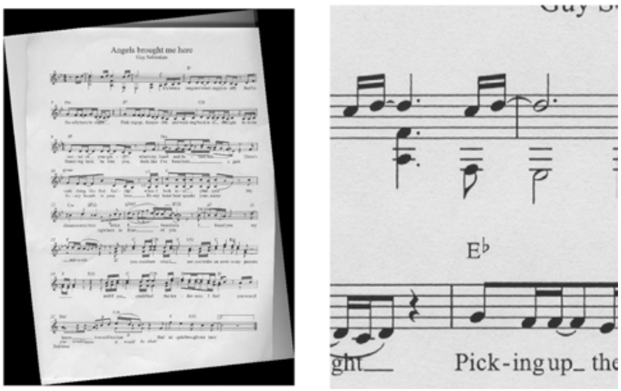
- There is a fainter and less well-defined tilted horizontal line, arising from the bar lines and note stems. This line is not at right angles to the FSC because the frequency units on the two axes are different.

[1] Throughout this paper we use x and y to denote the horizontal and vertical coordinates respectively in both the space domain and the frequency domain. In the spatial domain x and y measure distances in pixels, whereas in the frequency domain they measure spatial frequencies in cycles per image width/height.

- There is a very fine sharp horizontal line exactly along the frequency-space x-axis (only visible in enlarged versions of the image). This arises from a scanner artifact: a thin grey line that runs vertically down most of the left-hand edge of the image.

2.2 Removing a simple rotation

The simple test image shown above can be easily corrected by applying a rotation of $6.149°$. Figure 2a shows the whole page of music after applying that rotation, and Figure 2b shows a close up of a part of the page.



(a) (b)

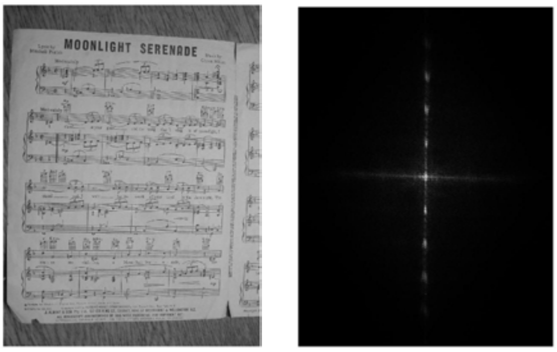
Figure 2: Test image one: (a) rotated by $6.149°$, and (b) a close up portion of it.

2.3 A perspective-distorted image

Figure 3a shows a much more difficult case: a 1536×2048 pixel digital camera image of a piece of sheet music. This example shows significant perspective distortion, and is not amenable to correction by a simple rotation of the whole image. There is also additional deformation introduced by the camera lens: notice the curvature on the left hand side of the music sheet, for example.

Figure 3b shows the Fourier transform of the perspective-distorted image. It can be seen that the FSC is now rather blurred, particularly well away from the origin, where it has a braided appearance. It is clear that while a "best fit" rotation angle could undoubtedly be found from the FFT and applied to the image, this is not going to be sufficient to correct the problems. A more general and flexible approach is required.

The blurriness of the FSC in Figure 3b arises because the various distortions, most notably perspective distortion, have resulted in different regions of the image being subject to different rotation angles. The FFT of small regions of the image will not suffer from this problem because the stave lines will be relatively straight and parallel over any small region. Hence we would expect to be able to obtain a good estimate of the image rotation angle over small regions of the image, particularly regions centred on stave lines. The rest of this paper addresses the following question: what information can we get by taking Fourier transforms of many small portions of an image, and how can we use that information to remove distortions in the image?



(a) (b)

Figure 3: Test image two, "Moonlight": (a) a perspective distorted music image and (b) its FFT.

2.4 Windowed Fourier transforms

The Fourier transform of a small portion of an image is called a *Windowed Fourier Transform*, because we can consider that the whole image has been weighted by a "window" function that is non-zero only near a single point of interest. The simplest window function is a "box" function that is unity within a rectangular region and zero outside that region. However, the use of box windows can result in "ringing" artifacts in the Fourier transform, consequently windows with a less sharp cutoff are preferred. The windowing function we have chosen is the *Hanning Window*, which uses a cosine function (between two troughs) to weight the points in the window.

Figure 4 shows the windowed Fourier transforms computed at a number of different places in the test image of Figure 3. It can be seen that the FFTs can be expected to yield good estimates of the local rotation angle whenever the window is positioned over the music staves, but might yield completely different angles elsewhere.

3 PERSPECTIVE DISTORTION

As an example of how the knowledge of local rotation angle(s) might be used in removing deformations, we focus on the problem of removing perspective distortion, such as that exhibited by Figure 3. Note that this treatment includes removal of pure image rotation as a simple subset. Throughout this section we assume that the "scene" being photographed is a rectangular page of music which we will refer to as the *source document*. This may be an isolated sheet of music or a page in a book of music. We call the photograph of the source document the *source image*. Our goal is to remove all geometric deformations including rotation from the source image to produce a new image that we call the *target image*. Unless stated otherwise we will also assume that the source document is planar.

The geometry of photography can, for an ideal camera, be modelled mathematically as a perspective projection. If the axis of the camera is perpendicular to the plane of the source document, the source image is a scaled and

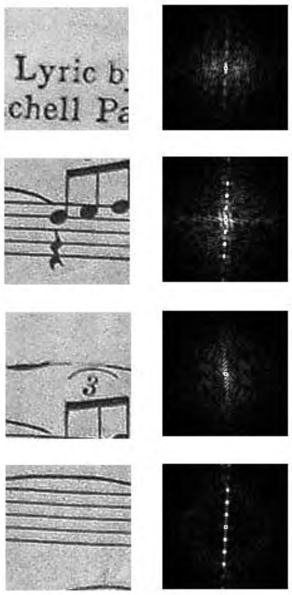


Figure 4: Some subimages and their (Hanning) windowed FFTs.

rotated but otherwise faithful reproduction of the document. In general, however, misalignment of the camera introduces a form of distortion called perspective distortion. This section discusses the properties of perspective-distorted images, provides a mathematical model for it, and develops a practical procedure for removing much of that distortion.

3.1 Properties of perspective-distorted images

Perspective projection is the mapping of points in a scene to points in an image by means of straight lines called projectors that pass through a common viewpoint called the centre of projection. For a given centre of projection C, each point P in the source document maps to a unique point P' in the source image, namely the point where the projector CP intersects the plane of projection, or *image plane*.

Under perspective projection straight lines project to straight lines, but angles between lines are altered by the projection process. In particular, lines that are parallel in the document are not parallel in the source image: all source lines parallel to a particular reference line in the scene converge in the source image to a single vanishing point. Any line in the document can be used as a reference line and has a corresponding vanishing point. The vanishing points associated with the x and y axes of the document are called they x and y-axis vanishing points respectively (see Figure 5), and the two together are called the *principal vanishing points*. We assume that the original document is rectangular and that the x and y axes are aligned with its boundaries.

General perspective projection maps the bounding rectangle of the document to an arbitrary quadrilateral in the source image. The boundaries of the quadrilateral converge on the principal vanishing points as shown in Figure 5.

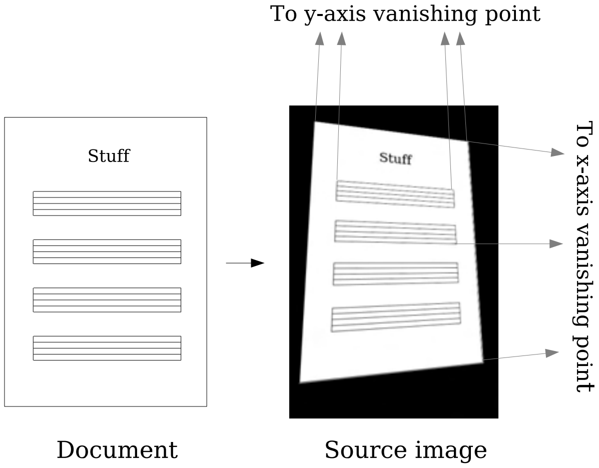


Figure 5: Test image three — an artifically constructed document and its perspective-distorted image.

3.2 Determining the perspective transformation

If we can identify in the source image the four corners of the document page being photographed, and if we also know the aspect ratio of the original document it is straightforward to compute an inverse transformation that will transform the entire page back to its initial rectangular shape. For example *MATLAB*, which we used for all the calculations in this paper, has a function *maketform* that computes the required transformation. The function takes as input the coordinates of the four corners of an arbitrary quadrilateral both before and after perspective projection (or after and before for the inverse transformation). In practice, however, identifying the document edges and corners in an image can be problematic, for the following reasons

- Some edges or corners of the document may be totally missing from the image, either because of inaccuracy in lining up the photograph, or to intentionally maximise the resolution of the image by not capturing the margins.

- Boundaries of documents can be ill-defined when photographed against a similar-coloured background.

- When photographing or scanning book pages, the bound edge may be curved or ill-defined as a consequence of the page being non-planar. These problems can also occur with loose pages, which may curl.

- The corners of the document photographed or scanned may be dog-eared, making the boundary non-rectangular, especially in older music that has had considerable use.

- Camera distortions, like pincushion and barrel distortion, give rise to non-straight edges.

Rather than using image-processing methods to determine the corners and edges of the original document within the source image, our method adopts a different approach to determining the perspective map by taking advantage of the stave lines that will be present in music images. Using a windowed FFT, we determine the local orientation of the principal axes at a large number of places throughout the image. From all that data we can determine best-fit estimates of the two principal vanishing points. Given the vanishing points we can then determine the perspective map. This approach is potentially more robust, immune to all the problems itemized above. We now explain this procedure in more detail.

3.3 Estimating local axis orientation

Our method for determining vanishing points requires estimates of the orientations of the document's x and y-axes at various points in the image. At each point of interest we take a windowed Fourier transform and attempt to estimate the rotation angle of both the x and y axes. As indicated by Figure 4, estimates of the x-axis rotation are excellent when the sample is positioned over the stave lines but can be poor elsewhere. Estimating the y-axis rotation is rather more problematic everywhere. We will return to the question of estimating y-axis rotation later, focusing on x-axis rotation to begin with.

Given a windowed Fourier transform we estimate the x-axis rotation by finding the maximum Fourier component "in the vicinity of" the Fourier y-axis, which is the axis onto which horizontal structures in the spatial domain map. More formally, we find the maximum Fourier component $F(x_i, y_i)$ where $y_i >= 25$ and $|\tan^{-1}\frac{x_i}{y_i}| < 20\pi/180$. This means we are confining our attention to spatial components with frequencies greater than 25 cycles per window width and with an orientation that is within 20° of horizontal.

The figure of 20° sets an arbitrary limit on the amount of rotation and/or distortion our method can handle. We confine our search to Fourier components whose frequency is greater than 25 cycles per transform window height in order to ensure reasonable accuracy in our estimate of the rotation angle, given the discretization errors.[2] We further reduce the effect of discretization errors by estimating the position of the maximum value to sub-pixel accuracy along a line perpendicular to the Fourier axis of interest.

In the case of the Fourier y-axis, if the maximum value is at location (x_i, y_i) with respect to the Fourier origin and has magnitude F_i, we find the magnitudes F_{i-1} and F_{i+1} of the components at (x_{i-1}, y_i) and (x_{i+1}, y_i). The magnitude F_{max} and position x_{max} of the maximum along the line $y = y_i$ is then estimated by fitting a tent function to the three values.[3] We use the value $\theta = \tan^{-1}\frac{x_{max}}{y_i}$ as our estimate of the rotation angle of the spatial x-axis at the centre of the Fourier window. Also, since strong spatial components like stave lines give rise to strong spatial frequencies, we use F_{max} as a confidence estimator for the subsequent determination of vanishing points.

Determining the local rotation angle of the spatial y-axis is exactly the same as for the x-axis but applying the function to the tranpose of the image. As we shall see later, the difference here is that y-direction lines are less common, and it can be a challenge to accurately identify the y-axis rotation.

3.4 Determining vanishing points in a perspective distorted image

Let us first consider the vanishing point for all lines parallel to the x-axis. Assume that we have estimates at various points in the image of the orientation of lines that were horizontal in the document, using the method of Section 3.3. We take samples using a regular grid approach, but to avoid aliasing problems[4] we "jitter" sample positions by randomly positioning each sample within a grid cell.

The goal is to find the coordinates of the vanishing point given such data. However, that is an ill-conditioned problem, since the lines are nearly parallel and the vanishing point is a long way from the image and may even, in the extreme case of a perfectly taken picture, be at infinity. So we instead specify the vanishing point in terms of two well-conditioned measures: the rotation angle θ_0 at the image origin and the *reciprocal*, α of the x-coordinate of the vanishing point.

With reference to Figure 6, assume the top left hand corner of the image is at location $(0, 0)$, with a coordinate system in which y increases downwards. Suppose that we can estimate from windowed Fourier transforms the angle $\theta(x, y)$ of the stave lines to the horizontal at an arbitrary set of points $\{(x, y)\}$ in the image. Assume all the horizontal lines meet at a vanishing point $V = (v_x, v_y)$. Then at each point $P = (x, y)$ for which we have a θ estimate:

$$tan\theta = \frac{v_y - y}{v_x - x} \tag{2}$$

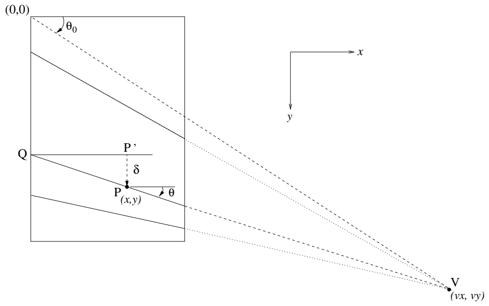


Figure 6: The vanishing point for "horizontal" lines.

Let $t = \tan\theta$ and $t_0 = \tan\theta_0 = v_y/v_x$. Eliminating v_y from Equation 2 and using $\alpha = \frac{1}{v_x}$ gives:

[2] Although the stave lines themselves probably have a spatial frequency less than 25 cycles across the window, they produce strong harmonics at higher frequencies.

[3] The tent function is $f(x) = max(0, abs(x - x_i))$.

[4] An extreme example of an aliasing problem would be if all our samples were to miss the stave lines because the vertical sample spacing was equal to the stave line spacing.

$$t = t_0 + \alpha(xt - y) \tag{3}$$

Equation 3 applies at each point P for which we have an estimate of θ. Hence, given a set of (x, y, θ) tuples, Equation 3 yields a set of equations, linear in the parameters t_0 and α, that can be solved in a least-squares sense for t_0 and α. We use a weighted least-squares method in which the Fourier magnitudes, F_{max} in Section 3.3, are used as the weights.

The y-axis vanishing point is found using the same approach.

3.5 Computing the perspective map from the vanishing points

Having determined the vanishing points for the source image, we now wish to find the transformation from the source image to the target image, which is an image that is a non-distorted non-rotated representation of the original document. At this stage we have an image and estimates of its two vanishing points. We do not know what portion of the image is occupied by the document, nor the aspect ratio of the document.

Our method is to use the vanishing points to construct a quadrilateral that is the perspective projection of an axis-aligned rectangle in the original document space. For small deformations, our quadrilateral occupies most of the image. We then compute the transformation that will map the quadrilateral to a rectangle in the target image with the same dimensions as the source image. This approach correctly restores all horizontal and vertical lines in the document but may introduce some aspect-ratio distortion. If the perspective distortion is small the aspect-ratio distortion is also small and is unlikely to be a significant problem with music images.

With reference to Figure 7, the steps in determining the perspective map given the two principal vanishing points P and Q are:

- Take the top left hand corner of the image as one of the quadrilateral vertices A.
- Determine vertices B and C, which are the points where the lines AP and AQ intersect the image boundaries.
- Find the vertex D where the lines CP and BQ intersect.

The vanishing points, computed using Equation 3, are in $(\tan\theta, \alpha)$ form. They are generally a long distance from the image and may even be at infinity. Hence the above geometric calculation should be done in a way that avoids explicit computation of the cartesian coordinates of the vanishing points. Our approach is to compute an algebraic solution, as follows.

In Figure 7, w and h are the width and height respectively of the source image, which has principal vanishing points P and Q. ABDC is the inscribed quadrilateral that will be mapped by the inverse perspective transformation to the boundary of the target image, as described above. Using a notation like that of Equation 3, P and Q are the Cartesian points $(1/\alpha, \tan\theta/\alpha)$ and $(1/\beta, \tan\phi/\beta)$.

B and C are the points $(w, w\tan\theta)$ and $(h\tan\phi, h)$ respectively. D is the point of intersection of the lines BQ and CP. Solving for D yields

$$\left(\frac{hw\alpha s - hs + hw\beta - w}{h\alpha w\beta - 1}, \frac{h\alpha w - h + wth\beta - wt}{h\alpha w\beta - 1}\right) \tag{4}$$

where $t = \tan\theta$ and $s = \tan\phi$. Note that this expression is well conditioned when the vanishing points are at infinity, i.e., when α and/or β are zero.

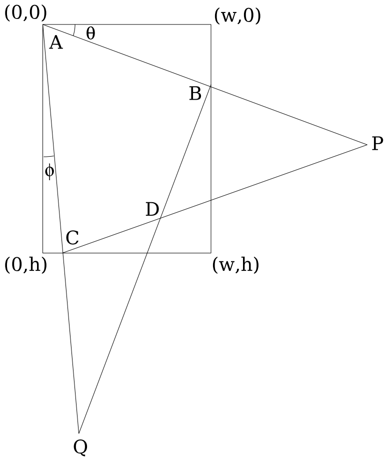


Figure 7: An image rectangle, its vanishing points, and the inscribed quadrilateral

4 RESULTS

In this section we evaluate the method of the previous section on three test images:

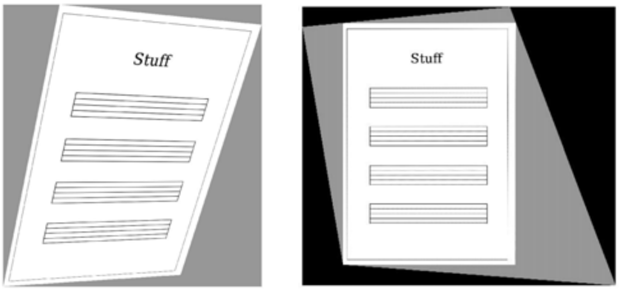
- A simple synthetic test document, shown on the left of Figure 5, which we subjected to a variety of rotations and perspective distortions.
- A test image (Figure 1) which has had negligible perspective distortion but significant rotation. We refer to this as the *Angels* test image.
- The motivating test image (Figure 3) which has had significant perspective distortion plus some nonlinear distortions, most noticeably barrel distortion. We call this the *Moonlight* test image.

4.1 The synthetic image, rotated

The synthetic test image was rotated by 7°. Orientations were estimated in a 5×7 grid, with one orientation estimation at a random position within each grid cell. A 100×100 window was used for the windowed FFT, so the cell grid is inset from the image boundary by 50 pixels (half the window size) all around. The original image is 1810×1396 pixels. As expected, good estimations of orientation were obtained only in the vicinity of strong vertical or horizontal spatial structures. The estimations of the vertical orientation are generally very weak except around the image boundary and the bar lines at the ends of the staves. Nonetheless, the best-fit estimations of both the horizontal and vertical vanishing points were generally very reliable because only the high-confidence estimates contributed significantly.

We ran 10 different trials on the test image with different random samplings each time. The worst result occurred when one of the samples was taken in the downward curve of the ‘S’ character in the title, returning a strong x-axis rotation estimate of $-14°$. With only 35 samples in total, and many of those having very low confidence, the single outlier had a significant effect on the x-axis vanishing point estimate. However, even with that large an error, the computed target image still looked very good; it showed only a very small amount of introduced perspective distortion along the x-axis. Errors of this sort could be reduced either by a more sophisticated fitting method that eliminated outliers or simply by increasing the number of samples. Using a larger window size for the FFT would also help: an optimal window size should accommodate a full set of stave lines and be appreciably larger than any text present. The window size of 100 pixels used for all the tests in this section is suitable for the other two test images but is a bit small for this synthetic test image, which has a large font and a wide set of stave lines.

We also created a perspective deformed synthetic image (see Figure 8a), and used our method to determine the vanishing points in 11 randomised trials. Figure 8b shows a typical target image. The recovery of the source document is essentially perfect apart from the fact that the image needs to be cropped.



(a) (b)

Figure 8: (a) The perspective-deformed synthetic test image, and (b) the target image recovered from the perspective-deformed synthetic test image.

4.2 Rotation and perspective distortion test images

We applied our algorithm to the genuine test image of Figure 1, which is an image with very little perspective distortion but significant rotation. The results of Section 2.1 suggest that the rotation angle is $6.149°$ so we would expect to get $x-$ and y-vanishing points of (0.1077,0) and (-0.1077,0) respectively.

In 10 randomised trials to determine the vanishing points we found that while the results for the x-axis vanishing point were consistently good, the estimates for the y-axis vanishing point were highly variable. Many of the y-axis vanishing points were seriously wrong and led to unsatisfactory target images in which the stave lines were horizontal but there was severe perspective distortion along the y-axis.

We tried increasing the sampling grid to 10 × 14, giving 140 samples in total, four times as many as before. However, poor results still occurred fairly frequently, and many of the highest-confidence orientation estimates were wrong. Particularly bad estimates arose from samples taken in the treble-clefs and in a portion of italicized text.

The *Moonlight* test image, which suffers from severe perspective distortion, yields similar results to the *Angels* image. Figure 9 shows the horizontal vectors that were detected in one trial. Again we obtained consistent estimation of the horizontal vanishing point (a variation of $\pm 7\%$ in $\tan\theta$ and $\pm 5\%$ in α), but much more erratic estimations of the vertical vanishing point.

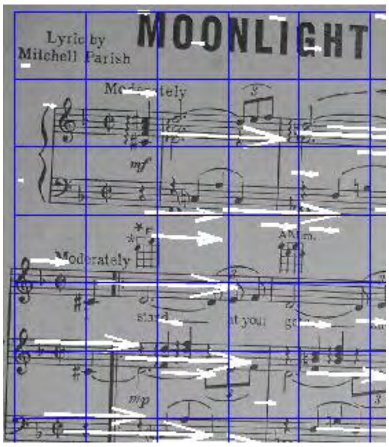


Figure 9: The vectors found by the FFT window of the “Moonlight” test.

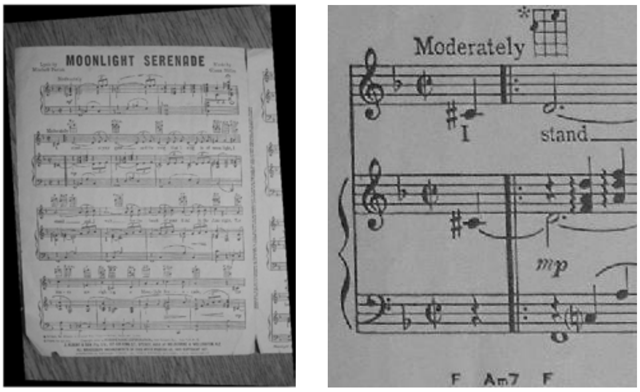
The music in the *Moonlight* test image was photographed on a wood-grain table and the grain of this table adds to the problems in determining vanishing points. When we restricted the sampling grid to the centre 80% of the image, thereby avoiding the woodgrain regions, the estimations of the x-axis vanishing point are slightly improved, though not in a statistically significant way, but the estimations of the y-axis vanishing point are significantly different from before and even more variable. If we use those vanishing points to “correct” for perspective distortion we tend to get results in which the stave lines are horizontal and parallel but the images show severe and indeed increased perspective distortion along the y-axis.

4.3 The best we can do

The above results show that we can obtain good results for the x-axis vanishing point. This allows us to correct images like the the Moonlight test in such a way that the stave lines are all horizontal. Subjectively this yields an enormous improvement. However, attempts to further correct for perspective distortion along the y axis have proved discouraging. While it might be possible to use sampled windowed FFTs to estimate local y-axis orientation, we believe that the sampling would have to be constrained, say, by a feature detection algorithm, if the method is to bear fruit.

Using only FFT-based techniques, the best compromise we have come up with is to use only the x-axis vanishing point to correct perspective distortion. We position the y-axis vanishing point at infinity on a line through the centre of the image, perpendicular to the line through the same centre to the x-axis vanishing point.

This approach works perfectly for rotated images like the Angels test image and allows us to at least obtain horizontal stave lines in cases like the Moonlight test image, as illustrated in Figure 10a. Figure 10b shows a close up portion of the corrected image.



(a) (b)

Figure 10: The result of applying our "best compromise" method to the Moonlight test image: a) full page and b) a closeup.

The experimental implementations were not designed for speed, and were implemented in Matlab. Processing of each page typically takes around 30 seconds, and there are likely to be many opportunities for speed improvement in an implementation.

5 CONCLUSIONS

Agile digital music libraries benefit from having fast and simple capture techniques, and this paper has illustrated a useful approach that uses a digital camera to capture music quickly, without the user having to be so concerned about introducing rotation or perspective distortion. Applying a windowed FFT to music is very effective at determining where the local x-axis is due to the predominance of stave lines. This leads to a reliable method that corrects for rotations and for x-axis perspective distortion. Stave lines in the corrected image are horizontal and free of obvious aliasing artifacts. However, determining the local y-axis orientation with the windowed FFT is more difficult because vertical lines are not so dominant. As a consequence, correcting y-axis perspective distortion is problematic. While this paper has made some progress towards a viable solution, a reliable algorithm will probably need to apply feature-extraction methods first, so that vertical orientation estimates can be made only in regions likely to yield good results. Further automation of the process would benefit from adaptive thresholding to allow for uneven lighting on the page, and automatic cropping to remove artifacts around the adjusted image.

REFERENCES

D. Bainbridge and T. C. Bell. The challenge of optical music recognition. *Computers and the Humanities*, 35 (2):95–121, May 2001.

S. Baumann and A. Dengel. Transforming printed piano music into MIDI. In H. Bunke, editor, *Advances in Structural and Syntactic Pattern Recognition (Proceedings of International Workshop on Structural and Syntactic Pattern Recognition)*, volume 5 of *Series in Machine Perception and Artificial Intelligence*, pages 363–372, Bern, 1992. World Scientific.

N. P. Carter. *Automatic Recognition of Printed Music in the Context of Electronic Publishing*. Ph.D. thesis, Departments of Physics and Music, University of Surrey, Guildford, UK, Feb. 1989.

A. T. Clarke, B. M. Brown, and M. P. Thorne. Inexpensive optical character recognition of music notation: A new alternative for publishers. In *Proceedings of the Computers in Music Research Conference*, pages 84–87, Lancaster, UK, Apr. 1988.

I. Fujinaga, B. Pennycook, and B. Alphonce. The optical music recognition project. *Computers in Music Research*, 3:139–142, 1991.

H. Kato and S. Inokuchi. A recognition system for printed piano music using musical knowledge and constraints. In *Proceedings of the International Association for Pattern Recognition Workshop on Syntactic and Structural Pattern Recognition*, pages 231–248, Murray Hill, New Jersey, June 1990.

P. Martin and C. Bellissant. Low-level analysis and recognition of music drawing images. In *Proceedings of First International Conference on Document Analysis*, volume 1, pages 417–425, Saint-Malo, France, 1991.

T. Matsushima, T. Harada, I. Sonomoto, K. Kanamori, A. Uesugi, Y. Nimura, S. Hashimoto, and S. Ohteru. Automated recognition system for musical score – the vision system of WABOT-2. In *Bulletin of Science and Engineering Research Laboratory*, volume 112, pages 25–52. Waseda University, Tokyo, Japan, Sept. 1985.

T. Reed. *Optical Music Recognition*. M.Sc. thesis, Department of Computer Science, University of Calgary, Canada, Sept. 1995.

J. Riley and I. Fujinaga. Recommended best practices for digital image capture of musical scores. *OCLC Systems and Services*, 19(2):62–69, 2003. ISSN 1065-075X.

M. Roth. An approach to recognition of printed music. M.Sc. thesis, Eidgenössische Technische Hochschule, Zürich, Jan. 1994.

M. J. Taylor, A. Zappala, W. M. Newman, and C. R. Dance. Documents through cameras. *Image Vision Comput.*, 17(11):831–844, 1999.