2004-01-01

Fast Registration of Tabular Document Images Using the Fourier-Mellin Transform

William A. Barrett
william_barrett@byu.edu

Luke A. D. Hutchison

Follow this and additional works at: https://scholarsarchive.byu.edu/facpub

Part of the Computer Sciences Commons

Original Publication Citation

BYU ScholarsArchive Citation
https://scholarsarchive.byu.edu/facpub/463

This Peer-Reviewed Article is brought to you for free and open access by BYU ScholarsArchive. It has been accepted for inclusion in All Faculty Publications by an authorized administrator of BYU ScholarsArchive. For more information, please contact scholarsarchive@byu.edu, ellen_amatangelo@byu.edu.
Fast Registration of Tabular Document Images
Using the Fourier-Mellin Transform

Luke A. D. Hutchison
Graphics, Vision and Image Processing Lab
Brigham Young University
Provo, Utah, USA
lukeh@email.byu.edu

William A. Barrett
Graphics, Vision and Image Processing Lab
Brigham Young University
Provo, Utah, USA
barrett@cs.byu.edu

Abstract

A new technique is presented for quickly identifying global affine transformations applied to tabular document images, and to correct for those transformations. This technique, based on the Fourier-Mellin transform, is used to register (align) a set of tabular documents to each other. Each component of the affine transform is handled separately, which dramatically reduces the total parameter space of the problem. This method is robust, and deals with all components of the affine transform in a uniform way. The Fourier-Mellin transform is also extended to handle shear, which can approximate a small amount of perspective distortion, and to not need Blackman windowing for document images. In order to limit registration to foreground pixels only, and to eliminate Fourier “edge effects”, a novel, locally adaptive foreground-background segmentation algorithm is introduced, based on the median filter. An original method is also presented for automatically obtaining blank document templates from a set of registered document images. Finally, image registration is demonstrated as a tool for compression of document images which share the same template.

1. Introduction

In many image processing and document archival fields, the ability to register, or align, related images to one another in the presence of various image transformations is important [3, 21]. In image registration, one image is stretched to align to another in its salient features, effectively undoing the relative transformation between the two images [4].

The need for image registration arises when a set of images share features which need to be correlated to make sense of the relationship between the images. Image registration is therefore important when dealing with a set of images that shares a common template, e.g. in the case of batches of document images written on tabular forms [24]. By first registering the images, more information is available for reliably determining which pixels in the image are part of the original document form (Figure 1), making table structure recognition [9, 44, 38] and content extraction [8] potentially more reliable.

2. Motivation

Many image collections in digital libraries contain batches of documents sharing an identical tabular form, such as census records. The information in these forms is often handwritten. Unfortunately, handwritten information is currently hard to automatically extract in a reliable way [39], so most extraction work for genealogy is done manually by volunteers. For example, in October 2002, The Church of Jesus Christ of Latter-day Saints released for free access on their www.familysearch.org web
Figure 2. Example of manual extraction of date field from hospital death records, for genealogical indexing. Screenshots are from the UDE (Universal Data Extractor) environment. The manually-designated extraction field does not always align well with all images. (Names have been changed.)

The above extraction project was accomplished by distributing batches of images to volunteer extractors on CD, paper, or sometimes microfilm, depending on the extractors’ preference. A program called UDE (Universal Data Extractor) was built for those extracting from CD (Figure 2). UDE allows users to create a grid overlaying the image that specifies where the table cells are. The program then highlights each cell in turn during data entry. The user also has a way to modify the grid by moving it or skewing it, because sometimes documents don’t line up with the template (Figure 2(b)). All this has to be done by hand currently. Automatic document image registration would streamline and simplify this process, by aligning each document with the template grid, allowing extractors to get on with the work of extraction.

Even if the table cells in the document image are to be read by computer rather than by a human (in the case of automatic indexing by OCR [12] or handwriting recognition [36, 28, 39]), it is useful to have the image straightened out so that the table cells become axis-aligned rectangles, and so that the scale of all the images is exactly the same. This means that the text recognition algorithms do not need to be able to cope with potentially significant sources of global image distortion between different images.

2.2. Motivation: Document Image Compression

The Church of Jesus Christ of Latter-day Saints currently has archived 2.3 million rolls of genealogical microfilm, with each roll containing an average of 1300 images. Around 60,000 new films are added each year. In order to better preserve these films, and to make them readily accessible over the Internet, one million of the rolls which are likely to be most useful to genealogists are currently being selected, and digitally scanned. The films are scanned at approximately 200 dpi (relative to the original page size) in 8-bit grayscale, and with page sizes ranging from 8.5”×11” to 17”×20”. Images are typically 2000×3000 to 4000×5000 pixels in size. Uncompressed TIFF images therefore range...
in size from 6-20MB, averaging around 13MB. The million selected rolls of microfilm alone would require approximately 17 Petabytes to store in an uncompressed format. This would fill approximately 140,000 120GB hard drives. Obviously, efficient image compression mechanisms are important to a digital library project of this scale.

Between one third and one half of the images in the 2.3 million roll collection were written on some sort of tabular form (parish registers, census records, death records etc.). Tabular records are often the most valuable to genealogists, because of their information-rich nature, thus they are more likely to be selected for the million-roll digitized collection. We can exploit the repetitive nature of form templates in a series of tabular documents, improving image compression ratios by predictive encoding. This technique has been successfully used before at a finer scale, for compression of machine-printed text, by identifying glyphs on the page and storing them symbolically [18]. Image registration would allow us to find the form template that is shared between a set of registered documents, and to just store it once for the entire set of images. Each image would then be stored as a residual difference from the template.

3. Related Work

The brute-force approach to image registration under arbitrary affine transformation involves a search (local or global) through up to six dimensions of transform space (scale: two dimensions; translation: two dimensions; rotation: two dimensions; and shear: a possible further one dimension). For images of any significant resolution, this search space is unreasonably large. The brute-force approach is thus never used for image registration, and various algorithms which exploit properties of the specific problem domain of image registration have been developed to reduce the total parameter space that needs to be searched.

The majority of previous work related to document image registration has been in skew detection and correction, e.g. correcting for rotational variation when a document is not axis-aligned on a scanner. Most skew detection algorithms do not easily extend to handle scale, translation or other transformations. However, as most of the body of document image registration literature deals with skew detection, it is described below. A description of full image registration techniques then follows in Section 3.5.

3.1. Skew Detection: Projection Histogram Methods

A common strategy for finding the skew (rotation) angle of a document is to find the angle at which a projection histogram through the document has the strongest and the best-separated peaks, as determined by some cost function.

Many algorithms for skew angle detection by projection histogram have been described in the literature. Postl [29, 30] describes a “simulated skew scan” method, where a cost function (the premium) is maximized while scanning through the document at a range of angles. Baird [2] reduces each connected component to a single point and builds projection profiles from the set of these points – this algorithm is designed to work well with machine-printed text where each character is separate, as opposed to large tables filled with handwriting, where almost the entire image may be a single connected component.

3.2. Skew Detection: Hough Transform Methods

The Hough transform [13] is another method for determining document rotation [20]. The standard Hough transform detects straight lines by mapping points in the $(x, y)$ domain to sinusoidal curves in the $(\rho, \theta)$ domain. Local maxima in the $(\rho, \theta)$ domain correspond to straight lines in the original image.

Amin [1] describes a method where the image is thresholded, and then groups of connected components are iteratively built, one scanline at a time. The Hough transform is applied to one connected component in each group to determine the skew angle. The algorithm is somewhat similar to the algorithm of Hinds et al. [16], who use run-lengths rather than connected components to choose points for Hough analysis. Using larger connected features such as this can reduce the computational complexity of the Hough transform [27], but such block-based data reduction methods often don’t work well for tabular documents with handwritten content, because so much of the document ends up being connected.

In general, the Hough transform can be computationally expensive and is sensitive to noise. There is always a trade-off in accuracy of the recovered parameters and the size of the accumulator array (and thus the computational complexity). Issues such as accumulator smoothing and accumulator granularity have to be addressed during implementation.

3.3. Skew Detection: Vectorization Methods

Cao et al. [5] have developed a method for skew detection by fitting a straight line to the baseline of connected components. Again, this is designed for machine print and is not suited to tabular documents containing handwriting.

Nielsen [24] has developed a system for alignment of tabular microfilm images, wherein a vector-based template is extracted from the document, representing the lines of the table upon which the document was written. A “mapping algorithm” is used to match one document’s vector tem-
plate to another. A consensus template is formed by combining several snapped grids together, making the algorithm more robust to noise. The production of the consensus template also allows for the transformation mapping one image to another to be found. This algorithm works well, but the author assumes that there is no major rotation of the document image away from axis alignment, because the algorithm relies on axis-aligned projection histograms to find table lines. The algorithm does allow for some variation in scale, by a brute-force search through a small neighborhood of the scale space.

3.4. Skew Detection: Other Methods

Some other various methods and features used for skew angle detection include gradient direction [34], eigenvectors of the covariance matrix [25, 35], texture directional evidence analysis [33] and mathematical morphology [22]. A particularly interesting (and generalized) approach makes use of the Wigner-Ville distribution [17].

3.5. Full Image Registration Methods

It should be noted again that document skew detection methods such as those described above are rarely designed to detect scale, translation and shear components of the transform that best maps one image to another. If such non-rotational transformations are present in a set of images, a separate algorithm is needed to correct for these components of the transform once skew angle has been detected and corrected for.

Garris and Grother [14] introduce a method for both rotational and translational registration, based on detecting linear features in forms. They demonstrate the efficacy of their method for a set of scanned forms. However, scale is not detected because it is not a consideration for scanned documents. Digital microfilm images are usually produced with a camera, and thus scale correction is a definite requirement.

Wolberg and Zokai [41] present a method for robustly registering images under rotation and scale variations, by warping an image in the image-space into log-polar form: in this form, changes of rotation and scale become simple shifts in $x$ and $y$. The system does not handle translation however, and the center for the log-polar warp has to be chosen reliably for registration to work at all.

Algorithms also exist for registration of images in a hierarchical “coarse-to-fine” fashion [45, 40]. Wolberg and Zokai’s method in [40] is particularly interesting, as it recovers the parameters of a perspective distortion by breaking down the images into tiles, and then finding the best affine fit between pairs of tiles in the two images to be registered. The perspective parameters are inferred by the piecewise-affine match between the image.

Image registration for general (non-affine) image warp correction may be performed by identifying common features between two images, and estimating the transform model that maps one image to the other [46, 4]. These methods can be complex (and invariant feature selection strategies must be chosen), but highly-generalized techniques like these should work well for specific problem subsets, such as tabular document registration.

4. The Fourier-Mellin Transform

Kuglin and Hines [19] originally proposed that “phase correlation” be used for translational registration in the frequency domain. Postl [29] described a Fourier-based method for skew detection, using a radial line integral through a 2D power spectrum to determine the rotation angle. De Castro and Morandi [7] presented a two-step process, first identifying the angle of rotation and then determining the translational shift. If the change of image scale is also present, the images can be registered by combining the log-polar mapping of the power spectrum (corresponding to the Fourier-Mellin transform) with phase correlation [6, 31, 10, 32].

The Fourier-Mellin transform is a method for determining the rotation, scale and translation that best maps one image to another (see [32, 37] for details). The algorithm makes use of the Fourier shift theorem to eliminate spatial translation from consideration while determining rotation and scale in the frequency domain. The frequency domain is radially sampled at exponential intervals to create a log-polar mapping of the frequency space, $(\ln \rho, \theta)$. In this mapping, rotation and scale are represented as simple linear shifts. One of the two remaining dimensions (rotation and scale) is then “collapsed” by projection, leaving a simple 1D correlation to determine the remaining transform.

The Fourier-Mellin transform is a powerful tool for registering images under simple combinations of scale, rotation and translation. It also has several weaknesses, which are analyzed in excellent detail by Stone and McGuire in [37].

One limitation for document image registration is that the Fourier-Mellin transform, in its usual form, doesn’t handle shear distortion (meaning it is not a solution for registration under arbitrary affine transformations). (See Figure 3(a).) Shear correction is useful for approximating a small amount of perspective distortion in photographed documents.

A further limitation of the Fourier-Mellin transform is that there can potentially be problems recovering rotation, because the operation of “wrapping” (induced by the periodicity of the DFT) doesn’t commute with rotation (Figure 3(b)). Thus, high-magnitude “wrapped” features in the frequency space can interfere with the log-polar transform.
The Fourier-Mellin Transform also requires the projection of the two dimensional $(\ln \rho, \theta)$ space down to one dimension in the final step. As with any 2D correlation problem, reduction to two 1D correlations of projection histograms can potentially create spurious matches. Perhaps most severe, however, are “edge effects” resulting from sharp, high-frequency transitions in image intensities where edges wrap. When this happens, axis-aligned radial features are created in the frequency domain, which can throw off rotation angle detection (Figure 3(c)). Usually, a Blackman window [37] is applied to the Fourier transform to try to curb some of these effects – a square subregion of the image is chosen before applying the Fourier transform, and frequencies that do not fall inside the incircle of the region are set to zero. Everything that is not zeroed is weighted as a Gaussian function of radius from the origin. Unfortunately, such windowing requires that we do not use the whole image for registration, making the algorithm potentially sensitive to subregion selection.

5. New method for document image registration as a specialization of the Fourier-Mellin Transform

In this paper we present a specialization of the Fourier-Mellin transform which exploits the properties of tabular document images to overcome most of the standard problems with the Fourier-Mellin transform. We demonstrate this algorithm as a general-purpose registration tool for tabular documents.

Many of the problems experienced with the Fourier-Mellin transform, in the context of document image registration, occur because of the coloring of the image background, i.e., the parts of the image we are not interested in registering (the gray paper color, the potential black border around a page, etc.). In order to accurately register document images, we really need to completely remove this document background. We introduce below a novel automatic background removal algorithm for document images.

Figure 3. Standard problems with the Fourier-Mellin transform
5.1. Preprocessing – automatic “locally-adaptive” background removal using the median filter

Typically, a document is written with dark ink on light paper, so many foreground-background segmentation algorithms attempt to find a segmentation threshold based on the histogram of graylevel intensities [26]. Better algorithms are locally-adaptive, choosing the segmentation threshold according to statistical properties of the histogram of intensities for a local neighborhood [23].

Our foreground-background segmentation algorithm is not really a thresholding algorithm at all, it is more a “background removal algorithm”. Simply described, we perform median filtering of a document image to remove the smaller foreground features (writing, lines etc.). We then subtract this “background image” from the original image, leaving just the foreground (Figure 4).

Significantly, this background removal method does not produce bitonal output, but leaves all of the subtle foreground variations intact. If a bitonal image is required, all that is required is a simple thresholding operation (Figure 4(d)), although the extra information available in the grayscale foreground image (Figure 4(c)) is valuable.

The method works because foreground pixels usually form much less than half the pixels in any reasonably-sized neighborhood in a document image. Of course the kernel size has to be appropriately chosen for each image. In general a kernel of radius 17 pixels was found to suffice for full-page document images of up to 1600x1200. The kernel size can scale linearly with the image size and achieve the same effect. The median filter preserves sharp grayscale transitions, but tends to round off corners of features in 2D space, so it pays not to make the kernel much bigger than needed; additionally, median filtering is time-consuming for large kernels. If the kernel is too small, then parts of the document that should be left as foreground will be absorbed as background.

Fast algorithms for finding the median (or the k-th largest element) exist [11, 15]. Median filtering for background removal can be sped up by working on a much lower-resolution version of the image, as described in Section 7.4.

When the median image is subtracted from the original image, if any pixel ends up negative, it is assumed to be light-colored noise (“salt noise”), and the difference is set to be zero rather than negative, because it is assumed that anything lighter than the background color has to be noise. The amount of “salt noise” in a neighborhood may be used as an estimate of the amount of dark-colored “pepper noise” in a neighborhood, so that everything within a noise tolerance is set to pure white.

Most importantly for the problem of the Fourier-Mellin transform, this form of background removal is able to distinguish between the black of the foreground (the handwriting or document form), and the black of the background (the border around a microfilm document image). Thus black borders are stripped out automatically, and different background grayscale levels at opposite edges of an image are normalized to white. Blackman windowing [37] is therefore not needed to eliminate the problem shown in Figure 3(c), and the entire image (not just an arbitrarily chosen windowed portion of it) can be used in the registration process. In this way, preprocessing with a median filter can improve registration accuracy dramatically for the Fourier-Mellin transform.

5.2. Recovery of rotation

Observe (e.g. in Figure 3) that groups of parallel lines in an image act as “wavefronts”, and thus produce a pattern of “delta functions”, or peaks in magnitude in the frequency domain, corresponding to the harmonics of the wavefronts. Rotation of the image produces a corresponding rotation of these radial patterns. The most periodic linear components of an image will therefore produce the strongest radial line integral over the power spectrum, at right angles to the linear component. The most periodic components of a tabular document are the lines forming the table itself. This makes it easy to find the angle of rotation and shear of the table – we just look for the two largest peaks in the (smoothed) radial projection histogram of the power spectrum. This is more robust for this problem domain than the general Fourier-Mellin case, where the 2D log-polar space is collapsed down into a 1D correlation problem, because no significant information that could prevent a spurious registration is lost in the projection. In addition, the intransitivity of wrapping and rotation (Figure 3(b)) does not adversely affect the correct selection of rotation angle.

5.3. Recovery of shear

Interestingly, by looking for the two most significant maxima in a radial projection histogram, we can now decouple the two image axes, since a rotation angle is returned for each. Thus we can separately determine the rotation angle (for the horizontal axis) and the shear angle from vertical (for the vertical axis). Recovery of shear is something that is not possible with the standard Fourier-Mellin transform, because the log-polar space is nonlinearly distorted in a shear operation (Figure 3(a)). It is made possible here by exploiting properties of this particular problem domain (specifically, the two strong periodic components of tabular document templates). Some care does need to be taken with sheared images, however, because if there is any non-affine perspective distortion, the radial pattern of maxima tends to
Figure 4. New automatic background removal algorithm based on the median filter, used as a pre-processing stage to the Fourier-Mellin transform. This algorithm can eliminate the problem shown in Figure 3(c).
spread, so the recovery of scale (as described below) may not be able to be performed accurately by sampling in a single straight line.

For rotation angles within $\pm 45^\circ$, the closest radial maximum to vertical is used as the rotation angle of the horizontal lines in the table. If registration over a greater angle range than that is required, then during the recovery of scale (below), the correct mapping of axes between the two images must be determined, by finding the assignment that produces the strongest total scale correlation.

The recovery of rotation and shear angles for a document image allows the image to be axis-aligned, without even considering other documents. The task of image registration, however, requires two or more images to be registered to one another. We arbitrarily choose the first image, de-rotate it / de-shear it, and then use this as our “reference image” to which the scale and translation of the other images are matched. The first image may not be the optimal choice for a scale and translation reference point, due to image noise, low image quality, or non-ideal size. A better system would use a consensus-based approach to ensure the reference image was not badly chosen.

### 5.4. Recovery of scale

Scale recovery happens in log-polar coordinates, as in the normal Fourier-Mellin transform. However, because we are dealing with tabular document images, only the strongly linear components of the image are important for registration. Thus determination of scale need only involve conversion to log-polar form of the Fourier magnitudes along the direction of the detected radial projection maxima. While a radial projection is performed, we do not lose potential correlation features by collapsing to one dimension in this manner, unlike the case with the projection of log-polar space in the standard Fourier-Mellin transform.

We then end up with two log-polar projection histograms, each one sampled at antilogarithmic intervals along one of the radial axes recovered during rotation and shear detection. The positive- and negative-frequency magnitudes are summed, weighted by the size of the current interval, and stored in a histogram. The logarithmic base can be chosen so as to yield any desired accuracy in scale determination. A base of 1.001 gives ample accuracy for a range of images.

The rest of scale recovery proceeds as normal for the Fourier-Mellin transform. In log-$r$ space, a translation by $\log(S)$ corresponds to a change of scale by a factor of $S$:

$$r' = r \cdot S$$  
(1)

$$\Rightarrow \log(r') = \log(r) + \log(S).$$  
(2)

Thus determination of scale reduces to a 1D correlation problem. It turns out that this works much better than trying to dilate one histogram to match the other in non-log space, because resampling a histogram in the frequency domain is problematic.

### 5.5. Recovery of translation

Now that rotation, shear and scale have been recovered, the transformation matrix which inverts these transforms can be easily determined, and the transformed image is rendered into a temporary image buffer. The image in this buffer now needs only to be aligned with the reference image using 2-D correlation, which may be performed again in the frequency domain, by pointwise multiplication with the complex conjugate.

Although this final correlation step is the simplest operation of the whole process, an important observation must be made: straight 2D correlation is not guaranteed to find the right correlation offset without a good algorithm (such as that presented in Section 5.1) for removal of background, and possible black borders around a page. Without this preprocessing step, changes in intensity across the background of the image can adversely affect the choice of a correlation offset.

### 5.6. Quadratic parameter fitting

Quadratic parameter fitting was used in the neighborhood of all recovered parameters to better achieve subpixel accuracy in the results.

### 5.7. Summary

Based on the shift theorem of the Fourier transform, and properties specific to the problem domain of tabular document image registration, we can “peel off” one translation component at a time (even more so than is possible with the standard Fourier-Mellin transform), dramatically reducing the total parameter space of the problem.

### 6. Results

#### 6.1. Pointwise mean

Figure 5 shows three datasets to which the registration process was applied. The first and second image for each dataset show the result of pointwise-averaging all the images in the dataset together before and after registration respectively. The registered versions of the images are well-aligned in the first and second instance; the third dataset was well-aligned horizontally, but not vertically, because the line spacing was irregular on some pages in the dataset (probably due to the process used to print the census forms). Figure 6(b) shows a closeup of the pointwise mean of a set
of registered images. Note that the original document images were transformed with the final registration transformation to produce these images (i.e. the versions with the backgrounds removed were not used).

6.2. Pointwise median

The third image in each dataset shows the pointwise median of the set of registered images. By using the median rather than the mean, we are able to recover a fairly clean blank document template. Figure 6(c) shows a closeup of a pointwise median image.

Some handwriting is still present in the pointwise median image. By taking a pointwise percentile value at a higher percentile than the median (i.e. the 50th percentile), more of this handwriting can be excluded, at the expense of template line strength (Figure 6(d)).

Median filter background removal can of course also be used to remove the background from the blank document template obtained by taking the pointwise median.

6.3. Accuracy of algorithm

Qualitatively, the images registered very well in almost all cases (including test cases not presented here). In order to quantitatively measure the accuracy of these methods, a single image was taken and rotated through various random (fractional) angles, and then separately scaled by various random fractional amounts. Each transformed image was then registered to the original, and then the detected trans-
formation parameter was compared to randomly chosen parameter.

Over ten trials, the mean absolute error for the angle of rotation and for the scale factor were measured. Results were as follows.

<table>
<thead>
<tr>
<th>Transformation</th>
<th>Measured Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rotation/Shear</td>
<td>0.01°</td>
</tr>
<tr>
<td>Scale</td>
<td>0.035%</td>
</tr>
<tr>
<td>Translation</td>
<td>Not determined</td>
</tr>
</tbody>
</table>

Recovery of rotation/shear and scale parameters was measured as being very accurate. The accuracy measured was actually significantly greater than the amount of warp that would be observed in a document image from almost any source, due to lens distortion, printing distortion, or mechanical lag. The scale calculations accuracy, for example, implies that a 3000-pixel wide image would have an average error in width of one pixel after scale correction.

For translation, the subpixel accuracy of correlation was not measured, but it should be of a similar order of fineness of fit to the other results (parabolic fit is surprisingly good at 2D subpixel correlation, for how simple it is). Subjective analysis indicates a good level of accuracy in the final step of recovering the translational offset.

6.4. Non-affine transforms

Often, document images have undergone non-affine transformations, such as the perspective transform, or a spherical transform due to lens warping. Some, but not all, of this type of distortion may be corrected for using a shear, which is handled by the algorithm as described.

In order to test how well the algorithm performed under non-affine conditions, digital photographs were taken of a document from various angles and distances (Figure 7). (This document was particularly effective for frequency-domain analysis because of its highly periodic nature.) Notice the ghosting at the edges of the registered image, particularly in the second dataset: significant perspective distortion was present in some of the images. However, the resulting image is registered about as well as it could be without handling perspective transformations too.

6.5. Run time

In one example run, ten images, 3045×2269 pixels in size, were processed on a 2.2GHz P4 machine. Fourier analysis was performed using “The Fastest Fourier Transform in the West” (www.fftw.org). Total time taken to register
(a) Dataset 1 (Camera no more than 15° from the surface normal)

(b) Dataset 1, Pointwise mean before and after document image registration

(c) Dataset 2 (Camera up to 30° from the surface normal)

(d) Dataset 2, Pointwise mean before and after document image registration

Figure 7. Results of image registration of two datasets under non-affine transform
the ten images was 8m54s (approximately 53 seconds per image). Around 90% of this time, however, was spent in the final determination of translation, since a 2-D correlation is required to find the translational offset. The runtime may be reduced, with increased chance of erroneous registration, by using two 1-D correlations of the projection histograms rather than one 2-D correlation. The measured time also did not include time to median-filter the images – median filtering can increase the time by a factor of two to three. However, as described in Section 7.4, the time taken to perform background removal could be dramatically lessened by working on a low-res version of the image, and interpolating when subtracting it from the image (the median filter can be smaller, too, for lower-resolution images). Runtime could also be dramatically decreased by working with images of approximately half these dimensions.

For comparison, a set of ten smaller images, 900×736 pixels in size, were registered together. Median filtering with a kernel of size 17 took 8 seconds per image; rotation, shear and scale together took about 1.5 seconds per image, and translation took just under 1 second per image to determine. All in all, the entire process took around 12 seconds per image.

Even in spite of these potential optimizations, it may seem that the presented algorithm is extremely slow compared to commonly-reported deskew algorithm statistics. However, this algorithm is doing substantially more than a plain deskew algorithm – it embodies a complete and accurate registration solution for tabular document images.

7. Discussion

7.1. Robustness

Most steps in this algorithm are very robust. The least robust part of the algorithm is in fact the simplest part: the final step to recover the translational offset. As noted previously, it is very important that a good thresholding algorithm be used to separate foreground from background, and that large areas of black (foreground color) surrounding the page are removed prior to this correlation step. Failure to do so can result in badly registered images.

Under purely-affine distortion, rotation and shear angles are virtually always recovered correctly. If there is any perspective distortion in the image, then the radial ridge of maxima in the frequency domain begins to spread, and although the skew angles will be chosen somewhere in the correct range, the linear log r histogram used for scale calculation may be determined wrong. Thus significant perspective transforms are currently not well handled as an affine approximation. This could be improved by sampling for scale over an angular window of the chosen rotation angle.

Also, most tabular documents have significant numbers of strong horizontal lines (for Roman alphabets), but a few do not have strong vertical lines. Thus the vertical scale can probably be trusted more than the horizontal scale. If the value of the horizontal scale is unrealistic based on the value of the vertical scale, the vertical scale could be used for both.

7.2. Considerations in recovering rotation

It is assumed that images within a single dataset to be registered are within ±45° of one another in orientation, thus in the determination of rotation the smallest rotation needed to align each image to the reference image is performed. If arbitrary rotations of documents are allowed, all four possible axis-alignments (i.e. 90° rotations) must be performed to see which gives the best correlation magnitude.

7.3. Image registration for predictive image compression

Image registration may yield advantages in document image compression. We obtained a document template using the median filter, as in Figure 5, and subtracted this template from each registered image (Figure 8). The resulting set of image, lacking the template, were approximately 14% smaller when compressed using bzip2 than the originals. In this way, image information which is shared by many pages of a document may be stored once and removed from each page which shares the information. This is an example of “predictive encoding” for image compression. Further gains can be found by improving the subtraction of the template from each image, by local snapping, or removal afterwards of small connected components.

An interesting area that we have begun to explore is the possibilities for image compression afforded by background removal. The background of an image makes up the majority of its uncompressed size, and a significant part of its compressed size, yet it contains little of informational value. The median filter-based automatic background removal algorithm presented in this paper allows the background to be removed reliably without discreet foreground/background segmentation (as is employed in DjVu/JBIG2), and the “holes” in the resulting background image are filled in automatically by the median filter. This could enabling a whole new class of image compression algorithms to be developed, which store the background at a much lower resolution than the foreground, if at all.

It should be noted that resampling a document image under an affine transform (as is done during registration) actually raises the entropy of the image, making it less com-
pressible. Adjusting contrast, thresholding, etc. can be used to reduce the entropy again after resampling.

7.4. Optimization

All but the smallest correlation operations can be performed faster by pointwise multiplication of the Fourier transform of one image with the complex conjugate of the Fourier transform of the other image, followed by finding the largest maximum in the inverse Fourier transform of the result. Fourier correlation was used in the implementation of this algorithm.

In addition, the final 2D correlation used to recover scale may be performed as two separate 1-D correlations of the projection histograms of the images (which is much faster, but has a greater chance of finding the wrong correlation offset).

Both median filtering and the Fourier transform algorithm are also highly parallelizable, allowing for significant speedups in hardware or on multiprocessing architectures. Also, many operations may be performed at a lower resolution for speed. For example, the median filter background removal process can be accomplished by working on a low-resolution version of the background with a smaller median kernel, and the result can then be scaled up using bilinear or bicubic interpolation before it is subtracted from the full-resolution image.

8. Future work

The current algorithm does a reasonable job of registering documents under some degree of non-affine transformation, but since many microfilm documents contain slight perspective or lens distortion, extending this technique to correct for these distortions would render it useful in a wider variety of situations.

A consensus-based approach could be investigated to ensure the reference image was not badly chosen.

Methods for quickly and automatically determining optimal median filter parameters for a specific image or set of images would be immensely useful, to ensure high image qualities and low running times.

Further research should also be performed into the use of image registration for document image compression.

9. Conclusions

We have presented an algorithm for fast registration of tabular document images, based on the Fourier-Mellin transform. This algorithm is robust and accurate in aligning tabular documents under affine transformations, and does a reasonable job of aligning tabular documents under non-affine transformations commonly found in document photographs. A novel automatic background-removal algorithm was introduced that enables many limitations of the Fourier-Mellin transform to be overcome (such as the need for Blackman windowing) for document images. The Fourier-Mellin transform was adapted to the specific problem domain of tabular image registration, and deals with all components of the affine transform in a uniform way in the frequency domain. The Fourier-Mellin transform is also extended to handle shear. Using a set of registered images, a novel method for generation of blank forms was presented.
and application of this technique to document image compression were discussed.

References


