Improved Legibility of Text for Multiprojector Tiled Displays

Philip Tuddenham and Peter Robinson
University of Cambridge Computer Laboratory
15 JJ Thomson Avenue, Cambridge CB3 0FD, UK
`<firstname>.<lastname>@cl.cam.ac.uk`

Abstract

Displaying small text on large multiprojector tiled displays is challenging. Problems arise because text is badly affected by the image-warping techniques that these displays apply to rectify projector misalignment. As a consequence, there has been little progress with important large-display applications that require small text, such as collaborative tutoring or web-browsing. In this paper we present a new warping technique designed to preserve crisp text, based on recent work by Hereld and Stevens. Our technique produces good results, free of artifacts, when used in today's multiprojector displays. We evaluate the legibility of our technique against conventional interpolation-based warping and find that users prefer our technique. We describe an efficient and reusable implementation, and show how the increased legibility has allowed us to investigate two new applications.

1. Introduction

Projected displays have advanced considerably over the past 15 years. Large displays on walls and tabletops have moved from research prototypes [8, 13] into widespread use. They offer benefits for a range of applications, from visualisations of complex structures to multi-user applications. However, displaying small text on these displays is still difficult. This problem has discouraged research into any applications that require a display to show detailed information. For example, there has been next to no investigation of large displays for collaborative editing of documents, remote tutoring, collaborative web browsing, and programming.

The significance of this gap in the literature should not be underestimated: research into large displays has so far shown promising results but, if we are truly to believe that large displays will play a significant role in the workplaces of tomorrow, then we must investigate the kinds of applications for which people use their desks, meeting rooms and notice-boards today.

The problem with small text arises when several projectors are configured in a tiled array to create a large high-resolution display. Small errors in the mechanical alignment mean that images may overlap, or be rotated, or suffer from keystone distortions. In order to create the illusion of a single contiguous large display, such systems must rectify these misalignments and distortions by applying small warps (for example, a stretch or a rotation) to images in the computer's framebuffers [12, 14, 7]. This warping is typically performed using bilinear interpolation offered by texture-mapping functions on commodity graphics hardware. Unfortunately this results in a significant degradation in high-contrast features such as small text. Figure 1(a) shows a typical display where a small disparity between the pixel resolution of the source data and the display leads to blurring. (Alternative display technologies, such as tiled arrays of LCD panels [6] have not gained wide support because of the wide seams between panels, and because it is more difficult to interact with such displays using styluses and bare hands).

Recently, Hereld and Stevens [5] proposed a novel image-warping technique that has been designed to preserve crisp text in multiprojector tiled displays. Their algorithm identifies high-contrast elements, such as lines of text, in the source image and copies them, without interpolation, directly from the source image over the top of the interpolated image. The technique successfully generates crisp images for very mild transformations. However, on closer examination, our reproduction of their technique produces distracting artifacts for the kinds of transformations commonly used in multiprojector tiled displays (Figure 1(b)) and also at boundaries between projectors (Figure 2(a)). Because of these limitations, the technique cannot be used to improve the legibility of text on multiprojector displays. Furthermore, the authors do not provide an efficient implementation.

In this paper we develop and evaluate a new technique for improving the legibility of text on multiprojector displays, and use it to improve text legibility in two new applications:

- We analyse the limitations of Hereld and Stevens' algorithm (Section 2).

- Based on this analysis, we develop a new algorithm for improving the legibility of projected text (Section 3). Our algorithm produces good results free of artifacts under the transformations used in today's multiprojector displays (Figures 1(c) and 1(d)). It also produces good results at boundaries between projectors (Figure 2(b)).

- We conduct a user study to evaluate the new technique against conventional interpolation-based warping and find that users prefer the new technique (Section 4).

- Finally, we present an efficient OpenGL implementation that exhibits good performance and can be applied to any Java Swing window. The improved legibility has allowed us to investigate two new applications, collaborative document reviewing and collaborative web browsing (Section 5).

2. Prior Work

Hereld and Stevens [5] make the point that bilinear interpolation, as employed in conventional multiprojector displays, compromises the crispness of text for the sake of positioning it to the accuracy of fractional pixels. They argue that transforming the text in this way is "neither necessary nor advantageous" and examine possible solutions. Their alternative approach processes the image as an "ocean" of warpable content within which there are "islands" of high-contrast unwarpable content (such as lines of text). Each island is "pinned" to the pixel grid by its centroid while the ocean is warped, producing a warped image in which the islands nevertheless appear crisp.

Their algorithm proceeds as follows to warp the source image into the frame buffer:

1. Create a binary image indicating islands of high contrast, by computing the contrast image of the source image and then thresholding. A morphological close operation merges closely-neighbouring islands.

2. Warp the entire source image into the frame buffer using bilinear interpolation.

3. Copy each high-contrast island, pixel by pixel, from the source image into the frame buffer, obscuring some of the warped ocean. Each island is positioned by applying the warp to its centroid coordinates.

Hereld and Stevens demonstrate their approach by applying a very mild transformation (a horizontal scale by factor 1.005). The algorithm successfully produces a warped image in which photographs and other low contrast features
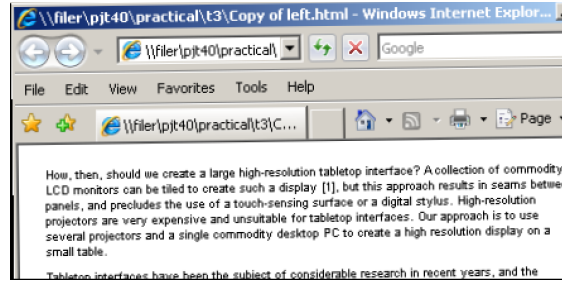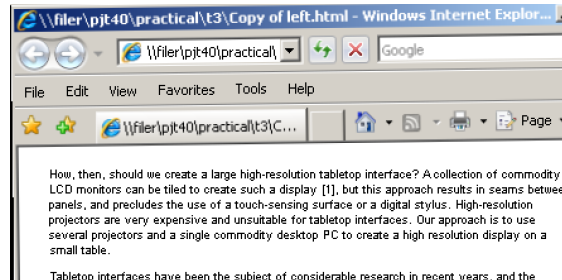

(a) Bilinear interpolation degrades high contrast features.
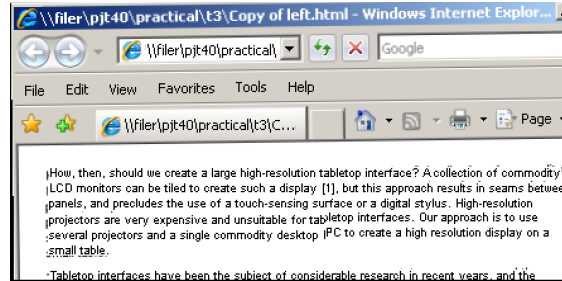
(b) Hereld and Stevens' technique produces crisp text but with distracting artifacts.
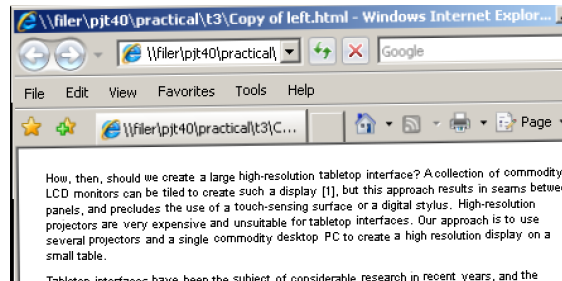
(c) Our words-as-islands technique produces crisp text without artifacts.

(d) Our lines-as-islands technique produces crisp text without artifacts.

Figure 1. A comparison of the warping techniques with a transformation likely to be found in a multiprojector display: a scale by factor 1.013 and a rotation by $0.75°$. This transformation would correct a disparity of 10px across a projector display of 1024px $\times$ 768px.

are warped using bilinear interpolation, while text and other high contrast features appear crisp.

However, in today's multiprojector displays it is usual to correct for pixel disparities of up to 10px across a projector display of 1024px × 768px [5], and also to correct for rotation and vertical scale, rather than just a horizontal scale. These conditions are equivalent to a horizontal and vertical scale by factor 1.013 and a rotation by 0.75°.

Under these more realistic conditions we find that our reproduction of their algorithm produces distracting artifacts (Figures 1(b) and 2(a)). Having analysed the algorithm, we found that these artifacts are caused by three problems:

1. The algorithm sometimes arbitrarily splits a single line of text into two islands or groups neighbouring lines into a single island, causing the line to appear split in the resulting image (for example, Figure 1(b) on the fifth line, between the words "desktop" and "PC").

2. When an island containing a line of text is copied pixel for pixel into the destination image, it fails to completely cover the warped line of text beneath and thus parts of the warped line are still visible at the edges. (For example, Figure 1(b) at the left-hand edge of the paragraph).

3. Their algorithm has no strategy for long islands (such of lines of text) that span two projectors. The algorithm aligns each island with the projector's pixels but, as described earlier, we expect projectors in a multiprojector tiled display to have slightly different pixel alignments because of keystone and small errors in mechanical alignment. A naive strategy therefore produces kinks in the lines of text and differences in text size at the boundary because of the misalignment (Figure 2(a)).

3. Improving Legibility

We propose a new algorithm that follows Hereld and Stevens' "islands in the ocean" approach while eliminating the three types of artifacts identified previously.
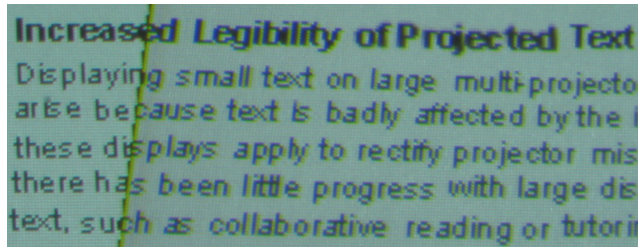
3.1. Words-as-Islands

Problem 1 stems from aiming to segment each line (sometimes arbitrary parts of various lines) as an individual island. Our algorithm addresses these problems by reliably segmenting each word as an individual island.

Many segmentation algorithms exist and are used in Optical Character Recognition systems [1]. However, these algorithms are designed to operate on text that has been printed and then scanned at a fairly high resolution, whereas we wish to operate quickly on text rendered at a much lower resolution by desktop applications. We follow Hereld and Stevens' approach to create a fast, pragmatic algorithm that reliably segments words.
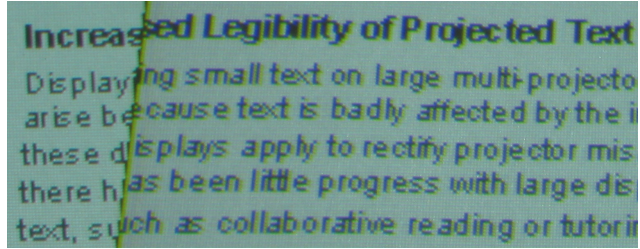
(a) Hereld and Stevens' technique produces distracting artifacts at projector boundaries.

(b) Our words-as-islands technique produces crisp text with few artifacts.

Figure 2. A comparison of the warping techniques at the boundary between two projectors using our tabletop display. The seam itself is noticable and could be minimised by blending between projectors; but nevertheless, the results stand.

We begin by creating a binary image that identifies the high-contrast pixels by using the standard contrast algorithm and then thresholding. Unfortunately, this labels not only the text itself as high contrast, but also the background pixels immediately bordering the text, and this sometimes merges neighbouring words into a single island. To solve this, we mark only those pixels which are high-contrast and also foreground-coloured, using a standard moving average to automatically infer whether the foreground colour is light or dark at each pixel. The combined algorithm reliably identifies high-contrast foreground pixels (Figure 3(a)).

We then merge neighbouring letters into words by performing a morphological dilation. To avoid inadvertently merging words with those above and below them, we dilate horizontally but not vertically.

Having segmented the words, we now use a standard edge-following algorithm to identify the bounding box of each word. Forcing the islands to be rectangular allows an efficient implementation (see Section 5) and does not affect visual performance. To avoid spurious results from images and icons that may appear alongside the text in web-pages and standard GUIs, we filter out rectangles with unlikely dimensions, and those whose background pixels are not of a consistent colour (we also record this colour for use later). This results in a set of rectangles that corresponds to the words in the source image (Figure 3(b)).
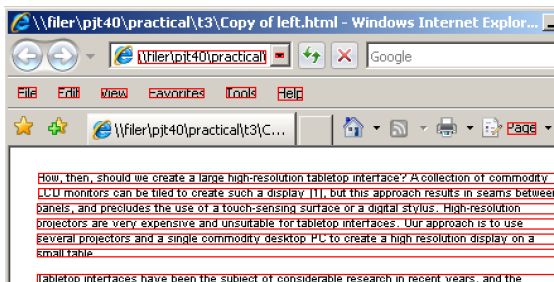
Following the "islands in the ocean" approach, we now warp the source image into the framebuffer using bilinear
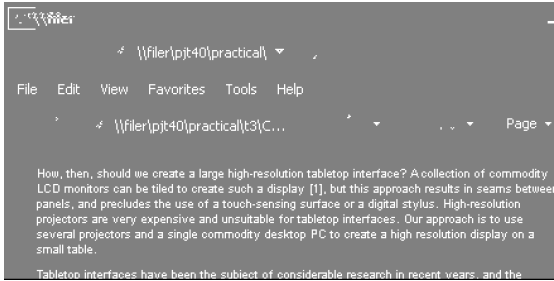
(a) Identification of high-contrast foreground pixels.
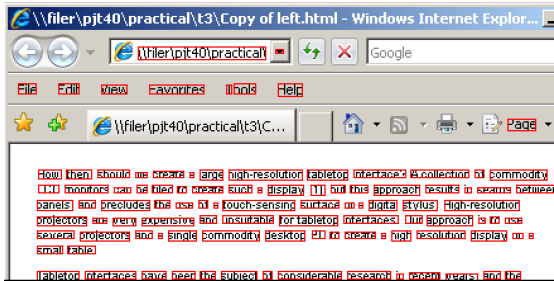
(b) Segmentation of individual words.

(c) Segmentation of individual lines of text.

Figure 3. Intermediate results from our techniques.

interpolation. We then draw filled polygons that completely cover the warped words in the framebuffer, using the background colours of the islands (identified previously). Finally, we copy each rectangular island unwarped, pixel for pixel, into the framebuffer, positioning it using the warped coordinates of its centroid. This middle polygon-filling stage addresses problem 2 of Hereld and Stevens' algorithm by ensuring that the warped words are not partially visible at the edges of the islands.

The result is a warped image without artifacts in which the text is crisper and more legible than under bilinear interpolation (Figure 1(c)). Focusing on reliable segmentation of individual words addresses not only problem 1, but also problem 3: the only islands that span two projectors are now individual words; and because words are relatively small, the disparity between the parts of the word projected on different projectors is small enough not to be overly distracting. Some small artifacts may be introduced when there are large errors in the mechanical alignment of the projectors,

but nevertheless the words appear crisp, and there are no unsightly kinks or changes in scale (Figure 2(b)). Although our implementation does not use masks to blend smoothly between projectors, but the principle is equally applicable to systems that do, provided the blend areas are relatively small.

3.2. Lines-as-Islands

The words-as-islands approach solves the problem of spanning two projectors, but sometimes changes the apparent word spacing and introduces a slight vertical disparity between words in the resulting image (apparent in the first line of Figure 1(c) between the words "high-resolution" and "tabletop"). While the resulting image is certainly more legible than that produced by bilinear interpolation, the effect is nevertheless slightly distracting.

For windows that do not span two projectors, we therefore propose an alternative algorithm that segments each line of text as an island and hence does not suffer this problem. We begin by segmenting words as islands, as described previously. We then group together islands that are horizontally within 8px of each other and also share the same background colour (identified previously). This simple algorithm is fast and reliable (Figure 3(c)): it does not arbitrarily split or group lines; and it ignores text with wide word spacing and where, therefore, the problems of the words-as-islands approach are not as apparent. The framebuffer results are pleasing (Figure 1(d)).

To improve text placement in the warped image, we finish by shifting the right-hand boundaries of the islands, to ensure that all islands within a paragraph share the same width (clearly visible in the line "small table" in Figure 3(c)). Our algorithm groups islands by their left coordinate and, for each group, sets all island widths to equal the maximum width in the group. We use each island's background colour to check that the area to its right is empty. Our tests indicate this algorithm is fast and robust.

3.3. Discussion

We have proposed two new algorithms: the words-as-islands technique for windows that straddle two projectors, and the lines-as-islands technique for those that do not. Both algorithms address the problems of Hereld and Stevens' algorithm, producing excellent results under transformations likely to be encountered in today's multiprojector displays (Figures 1(c) and 1(d)) and allow efficient implementations. We must now consider whether our algorithm is robust under different transformations.

Scaling with the words-as-islands technique alters the word spacing. Scaling down reduces the word spacing and is therefore undesirable, but can be avoided simply by rendering the source image at a slightly lower resolution so that the source image is always either scaled up or not scaled.

(a) Our words-as-islands technique with a scale by factor 1.05. Word spacing increases but this is not distracting.
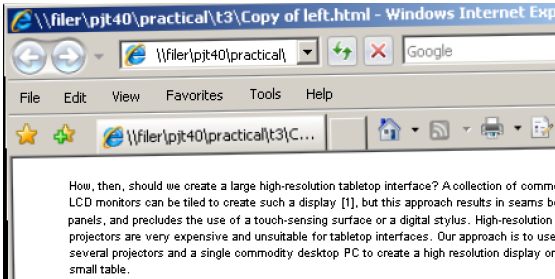
(b) Our lines-as-islands technique with a scale by factor 1.10 and a rotation by $1.0°$. The technique appears robust, even under this severe transformation.

Figure 4.

Scaling up increases the word spacing and our tests show that, at the smallest font sizes (where the effect is most apparent), a scale factor of 1.05 increases the word spacing by 50%. Even under this severe transformation, unlikely to be encountered in today's multiprojector displays, the results are good and the word spacing does not appear distracting (Figure 4(a)). This suggests that the technique could have applications not only to multiprojector displays, in which the scale is relatively mild, but also to keystone correction for single projectors, in which parts of the image are subject to more severe scaling.

Rotation using the words-as-islands technique results in slight vertical displacement between the words. Our tests show that, even at the smallest font sizes, rotations up to $0.75°$ result in at most an average of 1px vertical displacement every three words, which appears noticeable but not too distracting (Figure 1(c)). At more severe rotations the effect becomes distracting. Although our technique is therefore sensitive to rotation, it is nevertheless robust for the transformations likely to be encountered in today's multiprojector displays.

Note that, although we have considered only the effects of rotation and scaling, these results are not limited in scope. In the general case, the transformation is a planar homography [4]. But because words are reasonably small, the effects on an individual word are still well approximated by a translation, rotation and scale.

We can therefore conclude that the words-as-islands technique is robust for the transformations used in today's multiprojector displays, produces good results at severe scales, and is sensitive to severe rotations. By contrast, the lines-as-islands technique appears robust even under severe transformations (Figure 4(b)).

Finally, we note that systems such as Microsoft ClearType exploit sub-pixel display technology to increase text legibility. The resulting text has a lower contrast and hence is not reliably segmented by our algorithms. However, using our warping technique without ClearType nevertheless results in much more legible text than using conventional interpolation-based warping with ClearType.

4. User Study

Although our words-as-islands technique results in vertical displacement between words, our discussion suggests that it is nevertheless easier to read than text produced by interpolation, for the kind of warps commonly encountered in multiprojector displays. We tested this hypothesis with a user study. (By contrast, we saw no point in testing the lines-as-islands technique with a user study, since our results show that the appearance of each line is not altered).

Previous research [3, 11, 2] has investigated how text presentation affects reading, using a wide range of experiments and measures. They illustrate that subjective self-report measures are useful indicators of fatigue and legibility, and that reading time is not usually a useful measure in controlled experiments.

In our experiment, participants read a paragraph of projected text aloud and then indicated on a questionnaire the legibility of the text. In order to reduce the semantic aspects of the text, each paragraph consisted of 150 words randomly chosen from a set of 150 frequently used three and four letter words. We manipulated two independent variables:

1. Warping technique: either our pixel-aligned words-as-islands technique (P), or bilinear interpolation (I), or no warping at all (N).

2. The transformation level: either A (scale by 1.013, rotate by $0.75°$) or B (scale by 1.039, rotate by $0.75°$). We chose two transformations likely to be encountered in multiprojected displays, bearing in mind that in these displays, because of keystoning, scale is more of a problem than rotation.

4.1. Procedure

We used a counter-balanced within-subjects design. Each of the 14 subjects read a practice paragraph and then tested each of the five conditions. After each test, the participant was asked to indicate by marking a horizontal scale whether they found the text "easy or difficult to read".
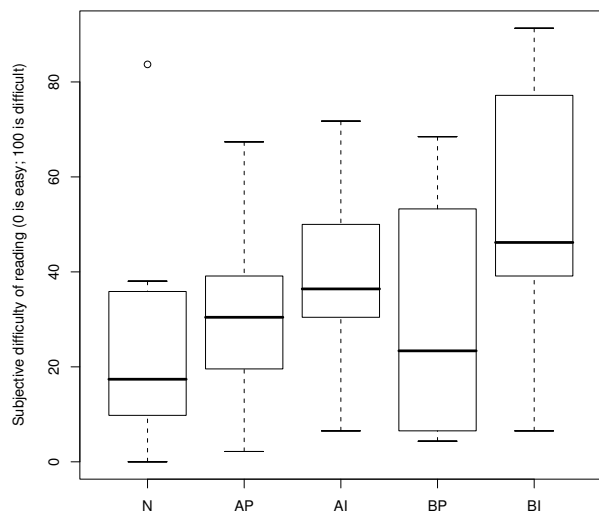
Figure 5. Subjective difficulty of reading at each condition. Transformations: A or B as described in the text. Warping technique: our pixel-aligned words-as-islands technique (P); bilinear interpolation (I); or no warping at all (N).

We rendered the text as black on white Arial font at the smallest font size at which the text could be rendered legibly (equivalent to 7pt font on a 96dpi screen), as this allows the maximum amount of information to be shown on a given display. However, when the warped text was projected, it appeared sufficiently large as not to limit legibility by its size. The warped text was projected using a single projector onto a light-grey horizontal surface at 45dpi under normal office lighting. The subjects were aged 20-30, spoke English as their first language and wore the glasses or lenses they would normally wear for reading.

4.2. Results

Figure 5 shows the subjective ease of reading for each of the conditions. A two-way repeated measures ANOVA (excluding the "no warping" condition) showed that participants believed that text rendered using our pixel-aligned warping technique was easier to read than text rendered using bilinear interpolation ($F(1,13) = 5.450$, $p < 0.05$). The transformation level had no significant effect and there was no significant interaction between warping technique and transformation.

4.3. Discussion

The results support our hypothesis that, in spite of the slight vertical displacement between words and the altered word spacing, text produced using our words-as-islands technique is still easier to read than text produced by interpolation, for the kind of warps commonly encountered in multiprojector displays.

As must be expected, the text that was not warped at all (N) was on average rated the easiest to read. Unfortunately this strategy does not address the need to compensate for small projector misalignments and so it cannot be used in today's multiprojector displays.

5. Implementation and Experiences

We have developed an efficient implementation of our algorithm by using conventional hardware. System performance is suitable for conventional multi-user large display applications, and the architecture allows it to be readily applied to any Java Swing application. We describe our implementation in more detail, and our experiences using it to improve the legibility of text in two tabletop multi-user applications.

5.1. Implementation

Our system design assumes that the application is similar to a conventional user interface:

- Content appears as rectangular tiles, such as web-pages, spreadsheets, dialog boxes or text documents.

- Multiple users sit around the table and wish to translate and rotate tiles of content on the warped display, for instance by dragging with a stylus, to pass them between each other, as they would with real artifacts on a real tabletop.

- Changes to the content of the tiles, such as a menu appearing or a button being pressed, are infrequent and often contained within a small area.

In our experience, these assumptions are valid across the vast majority of today's tabletop applications.

We use Java to implement the image-processing stages that identify the islands of high contrast using both the words-as-islands and the lines-as-islands approaches, and initially we run this procedure over the content of each tile.

Having now identified the islands of high contrast, we use OpenGL to exploit commodity graphics hardware to perform the rendering. Initially the unwarped tile image is loaded into texture memory. We then use the texture-mapping functions of the graphics hardware to warp the entire tile contents into the desired quadrilateral in the framebuffer using bilinear interpolation. Finally, we copy each rectangular island from the texture memory into the correct place in the framebuffer. If the tile is entirely within a single projector then we use the islands from the lines-as-islands technique, otherwise we use the words-as-islands technique. This second stage is extremely fast because we

are essentially copying memory from one buffer to another within the graphics card.

If the tile is to be translated or rotated on the display then we simply redraw the display using the graphics hardware as described above. There is no need to alter the texture memory or perform image processing, and thus the frame rate does not drop noticeably when using the new technique.

If the tile content changes (for example, a menu appears) then we must update the texture memory and re-identify the new high contrast islands. Processing can usually be restricted to the area that changed so this is relatively fast for small changes. Our image processing implementation takes 30ms to identify high contrast features in a 496px $\times$ 702px image of dense text, on a 3.2GHz Pentium 4 CPU. A reasonably-sized menu might therefore require about 5ms to process, and so the system performance is certainly suitable for conventional applications.

We have integrated this implementation into the freely-available TableTop Toolkit (T3) [10], a library for creating applications for multiprojector displays. The resulting system allows our pixel-aligned warping technique to be applied not only to specially-designed large display applications but also to almost any existing Java Swing window, from spreadsheets to web pages to dialog boxes.
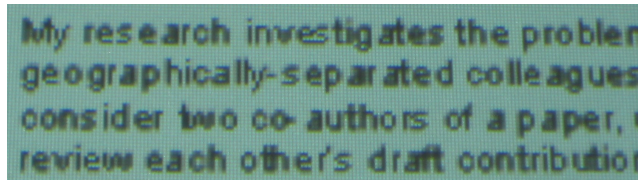
5.2. Experiences

Our tabletop uses 6 projectors in a tiled array to create a display with an area of approximately $0.5m^2$ and a resolution of around 50dpi. It uses a short calibration procedure to identify small transformations that it applies to correct for keystone and to compensate for small mechanical misalignments in the positioning of the projectors. Artifacts, such as web-pages, appear as rectangular tiles on the display and can be passed between different users sat around the table, by rotating and translating using a stylus.

We use our implementation to improve the legibility of text in two tabletop applications (Figure 6(a)). The first application is a collaborative web-browsing system that allows multiple users to interact with web pages on the table, pass them to each other and open new pages by clicking on the links. The second is a collaborative document review system in which multiple users navigate and annotate multi-page text documents and pass them between each other.

Both of our applications require the facility to display small legible text, in order that multiple projected pages can be viewed simultaneously by multiple collaborators, as with paper on a real table. We project the text to appear at around size 12pt (i.e. the size of 12pt text that has been printed onto A4 paper). The small transformations applied by the system use bilinear interpolation, with the effect of blurring the text. We successfully use our implementation of pixel-aligned warping, described above, to produce an image that is noticeably crisper and more legible (Figures 6(b)

(a) Collaborative web-browsing and document review applications.

(b) Conventional bilinear interpolation blurs the text in these applications.
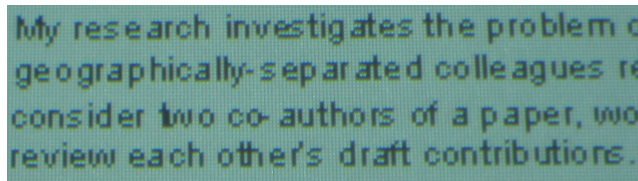
(c) Our technique produces crip text in these applications.

Figure 6. Experiences using our technique to improve the legibility of text on our 6-projector tabletop display.

and 6(c)).

In Section 3.3, we established that our technique is sensitive to severe rotation, in particular where the angle between the pixels of the tile and the pixels of the framebuffer exceeds $0.75°$ from some multiple of $90°$. In principle this presents no problems since the small transformations required for a multiprojector display can easily be kept within these bounds. However, recent tabletop research shows that the ability of users to rotate tabletop artifacts to arbitrary angles is a communication device in itself and is used, among other things, to establish ownership of artifacts and to attract attention [9]. In practice, therefore, we wish to allow free rotation of the artifacts to arbitrary angles.

We reconcile this problem by selectively applying our pixel-aligned warping technique only when the rotation is sufficiently mild; if not then we use bilinear interpolation. For users sitting around our square tabletop, we find that this compromise allows the benefits of free rotation, while still

providing improved legibility when the artifacts are directly facing one of the users. As noted previously, the transformation from tile to framebuffer is actually a planar homography, which is more general than an affine transform. In practice, we must therefore examine the rotation that the transform applies to each side of the tile rectangular tile.

6. Conclusions and Further Work

In spite of promising literature on the many virtues of large displays, applications that require small text have remained a largely unexplored research area. These applications are important because they reflect the tasks for which we currently use our desks and meeting room tables, and therefore there is great scope for large displays to offer a contribution. However, investigations are hindered because of the technical difficulties in legibly displaying small text on multiprojector displays.

In this paper we have built on initial work by Hereld and Stevens, to create an alternative warping technique for multiprojector displays that greatly improves the legibility of projected text. We have demonstrated that our technique produces crisp results, free of artifacts, under the transformations used in today's multiprojector displays. This is an improvement over Hereld and Stevens' technique and over conventional bilinear interpolation.

Our user study shows that, despite small changes to word positioning that arise from our technique, text is easier to read when rendered using our warping technique than when using conventional bilinear interpolation. We described an efficient and reusable implementation that uses commodity graphics hardware to achieve good performance and showed that it is not significantly slower than conventional bilinear interpolation for conventional applications. Finally, we showed how we have used the improved legibility to investigate two new tabletop applications.

Further incremental improvements to the algorithm may well produce better results at severe transformations for certain types of content, which would allow the technique to be used not only in multiprojector displays as we have done, but also in keystone correction and image scaling for single stand-alone projectors.

Acknowledgements

References

[1] M. L. Bernard, B. S. Chaparro, M. M. Mills, and C. G. Halcomb. A survey of methods and strategies in character segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 18(7):690–706, 1996.

[2] M. L. Bernard, B. S. Chaparro, M. M. Mills, and C. G. Halcomb. Comparing the effects of text size and format on the readability of computer-displayed times new roman and arial text. *Int. J. Hum.-Comput. Stud.*, 59(6):823–835, 2003.

[3] A. Dillon. Reading from paper versus screens: a critical review of the empirical literature. *Ergonomics*, 35:1297–1326, 1992.

[4] R. Hartley and A. Zisserman. *Multiple view geometry in computer vision*. Cambridge University Press, 2000.

[5] M. Hereld and R. Stevens. Pixel-aligned warping for multiprojector tiled displays. In *PROCAMS 2005*, 2005.

[6] C. Krumbholz, J. Leigh, A. Johnson, L. Renambot, and R. Kooima. Lambda table: High resolution tiled display table for interacting with large visualizations paper. In *Proc. Workshop on Advanced Collaborative Environments*, 2005.

[7] K. Li, H. Chen, Y. Chen, D. W. Clark, P. Cook, S. Damianakis, G. Essl, A. Finkelstein, T. Funkhouser, T. Housel, A. Klein, Z. Liu, E. Praun, R. Samanta, B. Shedd, J. P. Singh, G. Tzanetakis, and J. Zheng. Building and using a scalable display wall system. *IEEE Computer Graphics and Applications*, 20(4):29–37, 2000.

[8] E. R. Pedersen, K. McCall, T. P. Moran, and F. G. Halasz. Tivoli: an electronic whiteboard for informal workgroup meetings. In *CHI '93: Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 391–398, 1993.

[9] S. D. Scott, M. Sheelagh, T. Carpendale, and K. M. Inkpen. Territoriality in collaborative tabletop workspaces. In *Proceedings of the 2004 ACM conference on Computer supported cooperative work*, pages 294–303, 2004.

[10] P. Tuddenham and P. Robinson. T3: A toolkit for high-resolution tabletop interfaces. In *Ext. Abstr. 2006 ACM conference on Computer supported cooperative work*, 2006.

[11] R. A. Tyrrell, T. B. Pasquale, , T. Aten, and E. L. Francis. Empirical evaluation of user responses to reading text rendered using cleartype™ technologies. *SID Symposium Digest of Technical Papers*, 32(1):1205–1207, 2001.

[12] G. Wallace, O. J. Anshus, P. Bi, H. Chen, Y. Chen, D. Clark, P. Cook, A. Finkelstein, T. Funkhouser, A. Gupta, M. Hibbs, K. Li, Z. Liu, R. Samanta, R. Sukthankar, and O. Troyanskaya. Tools and applications for large-scale display walls. *IEEE Computer Graphics and Applications*, 25(4):24–33, 2005.

[13] P. Wellner. Interacting with paper on the digitaldesk. *Commun. ACM*, 36(7):87–96, 1993.

[14] R. Yang, D. Gotz, J. Hensley, H. Towles, and M. S. Brown. Pixelflex: a reconfigurable multi-projector display system. In *VIS '01: Proceedings of the conference on Visualization '01*, pages 167–174. IEEE Computer Society, 2001.