**SCHMIDT, D — TILT**

<6 images (3 in one figure)>

**TILT 2: Text to Image Linking Tool**

Schmidt, D.

TILT is a Web-service and graphical user interface designed to make it easier to study printed books or handwritten manuscripts. TILT recognises word-shapes in page images, then links them to words in an already existing transcription. Unlike a traditional OCR program, no actual characters are recognised. Instead, the links are used to provide visual cues to the user by highlighting corresponding parts of the image and the text as the user moves the mouse over the words or taps them.

Text-to-image linking was ﬁrst explored by Seales et al. (2000) in electronic editions of old English texts at the Uni­versity of Kentucky. Their technique was to manually draw mostly rectan­gles around areas of manuscripts and then link them to the text by means of embedded markup in the transcription (Dekhtyar et al., 2005), developing eventually the commercial EPT tool. Other examples include the British Museum’s electronic *Codex Sinaticus* (BM, 2009) and, using a different display technique, the ‘zoom topographic’ view of the Samuel Beckett digital manuscript project (Hulle et al., 2011), the TILE project (Reside et al., 2011), and the TextGrid text-to-image link editor (Al-Hajj and Küster, 2013). However, all of these methods were predominantly manual. Rather than automatically processing the text and images, they invite the user to manually draw shapes (rectangles, ovals, and polygons) around words, and to link them one at a time to words in the text. This is very tedious, and the linking methods also lead to problems of overlapping markup that can only be resolved by creating two separate transcriptions: one to contain the text-to-image links and one the formatted text (Middell and Wissenbach, 2011).

The earlier TILT tool (Schmidt, 2013) attempted to address the automa­tion of the text-to-image links by allowing words to be outlined via a single click, and for entire pages to have their shapes recognised and *linked* to the corresponding words of the transcription largely automatically. But TILT 1 proved to be limited in the way it could deliver a user interface. The key problem was that in order to automate the linking, it would be neces­sary to have access to a fully featured suite of image-processing tools. For this, the Java class library was chosen. However, the only way to mediate the automation process was to present the tool as a Java applet, which is notoriously difficult to run in a web context.

TILT 2 attempts to address this problem by separating the front-end GUI (link editing and management of the recognition process) from the back-end (page recognition) part. The back-end is accessible via a test interface (Schmidt, 2014), and is designed to act as a faceless service usable by anyone who has a page image and a transcription, and wants to link the two. The front-end GUI is currently under development and contains tools for revising the automatically generated word-shapes and links using standard web technologies.

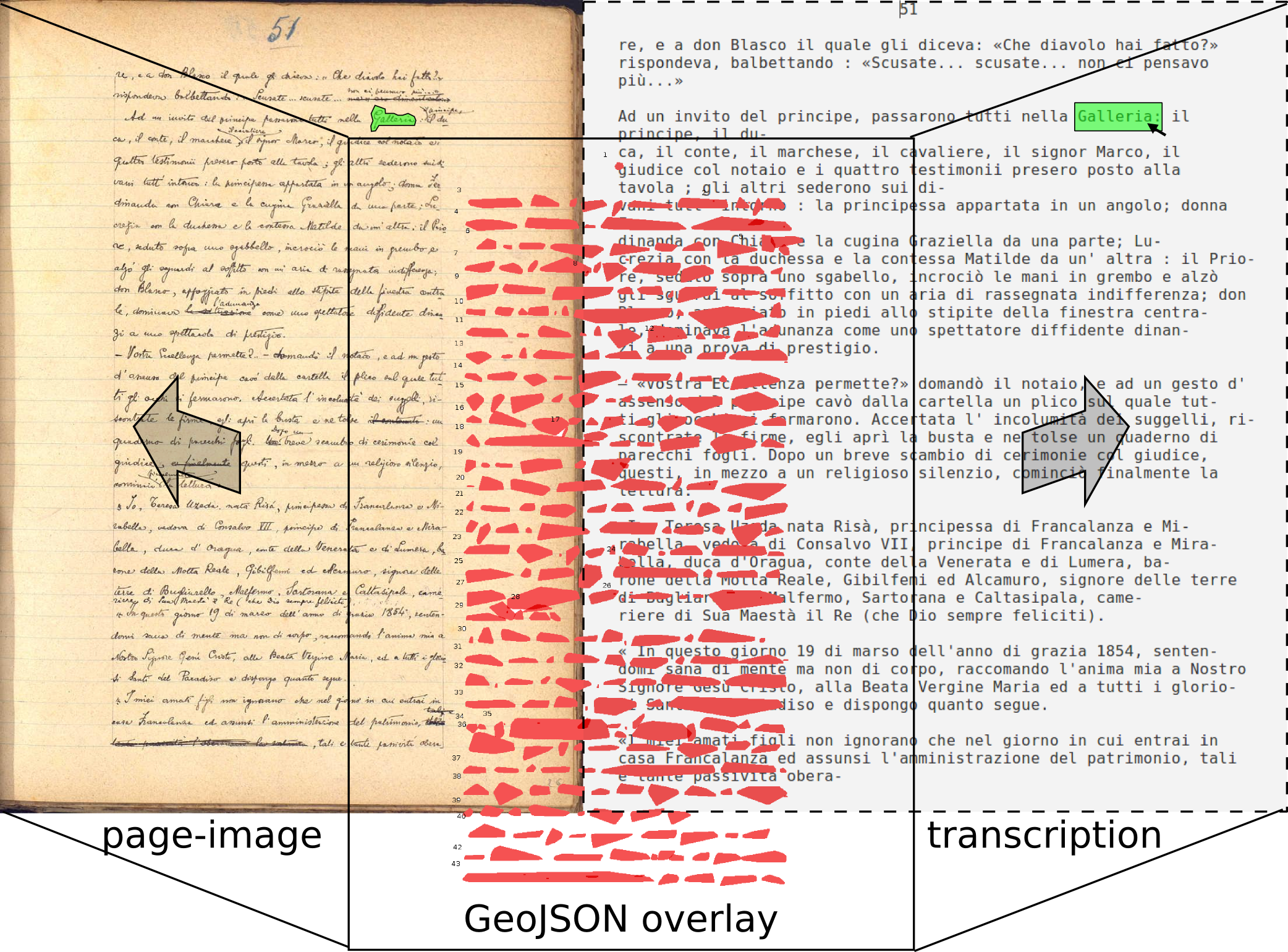


Figure 1. Text-to-image links and polygons as invisible GeoJSON overlay.

Rather than use embedded markup in the transcriptions to link word-shapes with the corresponding words in the text, TILT uses GeoJSON (Butler et al., 2014) to describe the page and its recognised shapes, and annotates it with the positions of the words in the text. As shown in Figure 1, both the text-to-image links and the polygons that overlay the image are wholly contained in an invisible GeoJSON doc­ument, which can be loaded from a database or generated if required. The transcription can be in HTML or plain text formats.

**Page Recognition**

Given the immense variety of manuscript types, the TILT service is divided into clear stages. Each stage feeds the result into later stages, ending in the generation of the word-level links. The stages are:

1. Conversion to greyscale.

2. Conversion to two-tone.

3. Removal of borders.

4. Line recognition.

5. Word recognition.

6. Linking word-shapes to actual words.

Of these, the first three conform closely to the standard processing steps of OCR and use techniques in the Java class library or the Ocropus toolset (Google, 2014).

The fourth step, however, represents a departure from standard OCR techniques. In order to recognise words, they must first be organised into lines. In printed books, recognition of lines is relatively easy. OCR programs use heuristics to determine text blocks and assume even line spacing. Skewed or warped lines must first be deskewed or dewarped. Since lines may curve or tilt naturally in manuscripts, a general line-recognition algorithm is needed that can cope with both printed and handwritten material. TILT’s approach is to first divide the image into narrow vertical strips. The strips are then reduced to a single average column of pixels, which is smoothed via a moving average algorithm. The peaks in the black pixels of each strip then are taken to correspond to lines. Finally, the peaks are linked up across the page using a greedy approach, by linking the closest points in each pair of adjacent columns first, then progressively more distant points, so long as this does not cause the lines to cross (Figure 2) (BL, 2014).

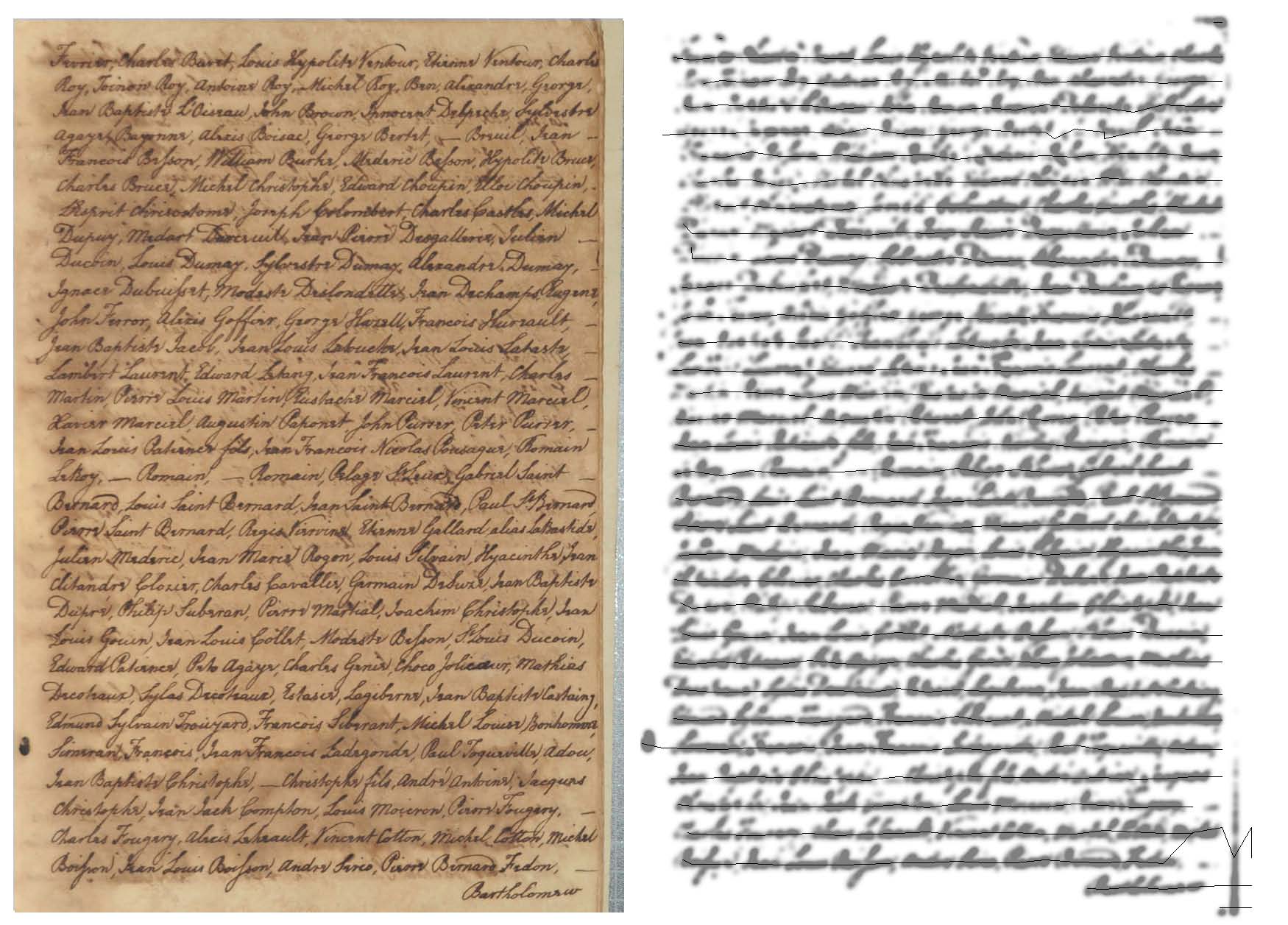


Figure 2. Line-recognition (Court of Oyer and Terminer for Trial of Attained Traitors record book [1796] from Grenada).

Once lines have been recognised they can be subdivided into words. (This assumes that texts *have* word divisions, which may not be the case in certain languages.) The current method uses the two-tone image to identify ‘blobs’ of connected pixels in an image, then surrounds them with a polygon, and finally measures the gaps between polygons (Stamatopoulos et al., 2010). So on a page with 230 words on 30 lines, the number of inter-word gaps to be expected would be 229−29=200. So, assuming that the 200 largest gaps are inter-word gaps, polygons not separated in this way can be merged to form word-shapes. Although this works well for printed texts, with success rates often as high as 98 to 100%, variation in the use of spacing in manuscripts limits its success. An alternative technique (Manmatha and Srimal, 1999) tries to determine an appropriate Gaussian blur to join up blobs into the expected number of words, and then split words on that basis, achieving 86% average word recognition on the test documents. Application of this technique to TILT2 is planned but not yet implemented.

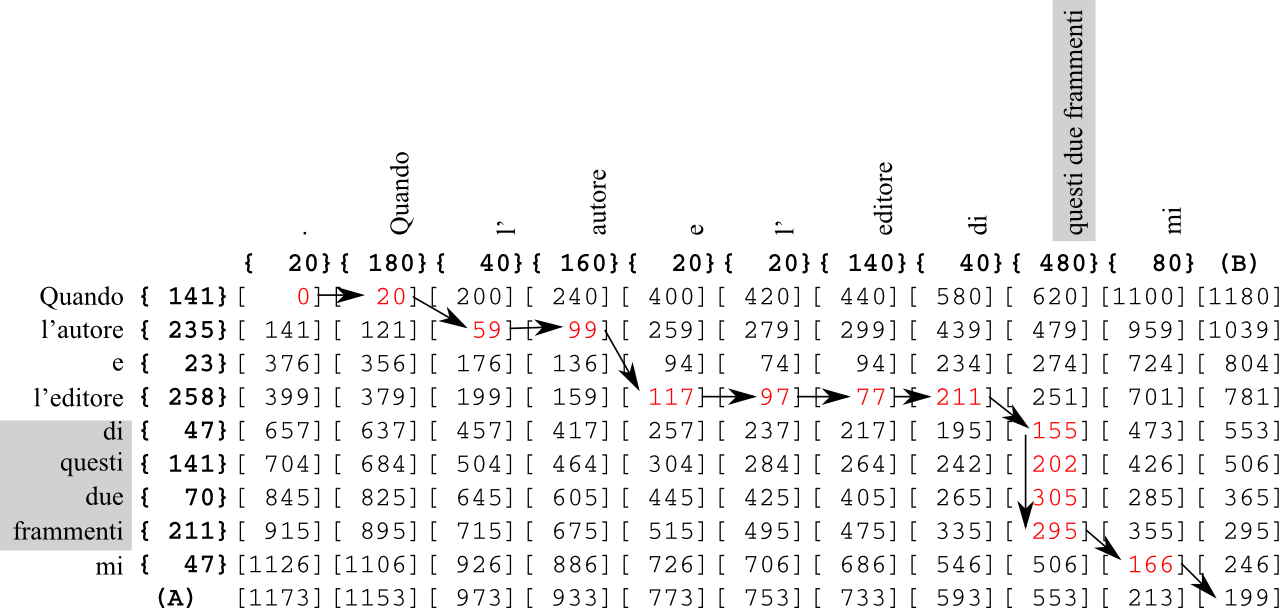


Figure 3. Alignment of word-widths in text and image.

The final linking step is actually the most reliable part of TILT. Since the numbers of word-shapes and words are now known, the approximate width of a single letter can be calculated. This allows the estimation of the pixel-widths of words in the text. Since the sequence of word-shapes in the image must match the sequence of word-shapes in the transcription, the two can be aligned by modifying a classic diff algorithm, such as Ukkonen’s (Ukkonen, 1985). But instead of matching letters in two texts, TILT matches word-widths and allows either several word-shapes in the page image to match one word of the transcription or several words in the transcription to match one word-shape in the image, as shown in Figure 3. This can result in polygons being split or merged to correct the alignment.

### Editing Interface

The editing interface aims to visualise the automatically generated text-to-image links and allow the user to adjust them manually. This uses a number of semi-automatic techniques to speed up the editing process (Figure 4):

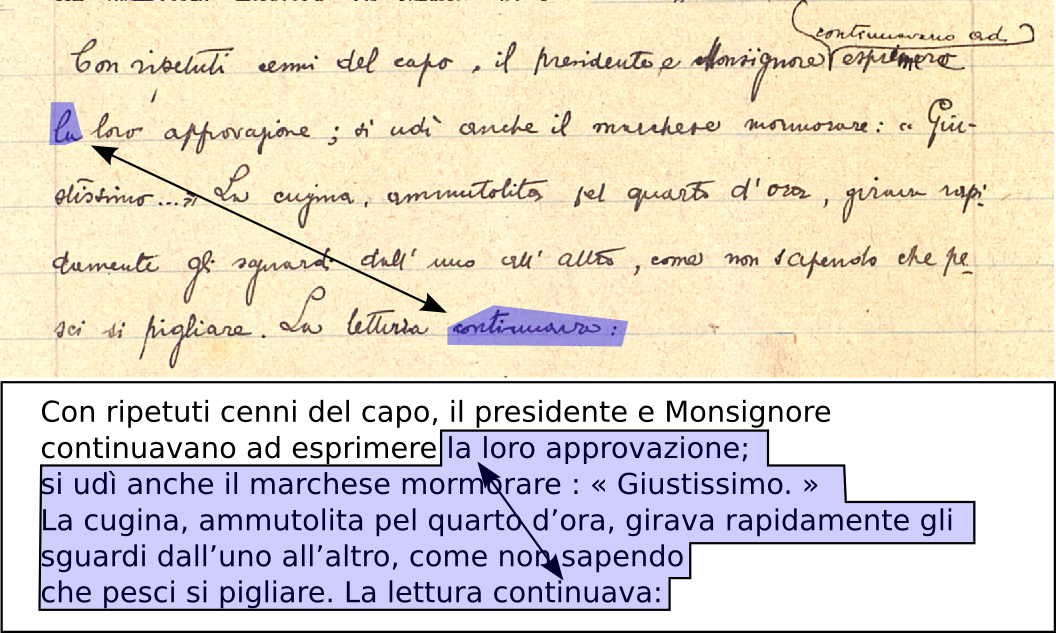
1. Redistributing words among the word-shapes between any two manually specified anchor points.

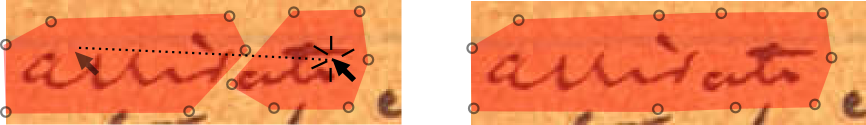
2. Merging two selected polygons when a word-shape has been incorrectly split by dragging across the shapes.

3. Splitting of polygonal shapes by dragging a line across merged words.

In addition to these techniques, manually adding and deleting points, and adjusting shapes provide a fallback when automatic methods fail.

Although currently incomplete, the editing interface is anticipated to be ready for demonstration at the conference (Schmidt, 2015).





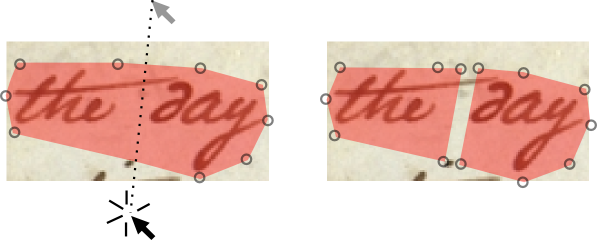


Figure 4: Semi-automatic editing tools: realignment with anchors (top), merging a split word (middle), splitting two merged words (bottom).

**Acknowledgements**

The TILT2 tool was one of two winners of the British Library Labs Competition 2014. The Center for Biodiversity Informatics at the Missouri Botanical Garden is currently supporting further development of the tool.

**References**

**Al-Hajj, Y. A. A. and Kuster, M. W.** (2013). The Text-Image-Link-Editor: A Tool for Linking Facsimiles and Transcriptions, and Image Annotations. *Literary and Linguistic Computing,* **28**(2): 190–98.

**BL.** (2014). Endangered Archives: EAP295/2: Collection of Court Records Held by the Grenada Supreme Court Registry [1765–1797]. http://eap.bl.uk/database/overview\_item.a4d?catId=140831;r=21726.

**BM.** (2009). Codex Sinaticus. http://www.codexsinaiticus.org/en/manuscript.aspx.

**Butler, H., Daly, M., Doyle, A., Gillies, S., Schaub, T. and Schmidt, C.** (2014). Geojson. http://geojson.org/.

**Dekhtyar, A., Iacob, I. E., Jaromczyk, J. W., Kiernan, K., Moore, N. and Porter, D. C.** (2005). Support for XML Markup of Image-Based Electronic Editions. *International Journal on Digital Libraries,* **6**(1) (February): 55–69*.*

**Google.** (2014). The OCRopus(tm) Open Source Document Analysis and OCR System. https://code.google.com/p/ocropus/.

**Manmatha, R. and Srimal, N.** (1999). Scale Space Technique for Word Segmentation in Handwritten Documents. In Nielsen, M., Johansen, P., Olsen, O. and Wieckert, J. (eds), *Scale-Space Theories in Computer Vision*. *Lecture Notes in Computer Science*. Vol. 1682. Springer Berlin Heidelberg, pp. 22–33.

**Middell, G. and Wissenbach, M.** (2011). Faust: Multiple Encodings and Diplomatic Transcript Layout. In *Philology in the Digital Age Annual TEI Conference*, Wurzburg, 10–16 October 2011, http://www.tei-c.org/Vault/MembersMeetings/2011/ﬁleadmin /04100700/\_temp\_/TEI­\_Book\_of\_Abstracts.pdf, p. 29.

**Reside, D., Lester, D., Porter, D. and Walsh, J.** (2011). Tile–Text Image Linking Environment. http://mith.umd.edu/tile/.

**Schmidt, D.** (2013). Text to Image Linking Tool (TILT). In *Digital Humanities 2013 Proceedings*, University of Nebraska, Lincoln, 16–19 July 2013, http://dh2013.unl.edu/abstracts/ab-112.html.

**Schmidt, D.** (2014). TILT2 Test Interface. http://ecdosis.net/tilt/test/post.

**Schmidt, D.** (2015). TILT2. http://www.github.com/AustESE-Infrastructure/TILT2.

**Seales, W. B., Griﬃoen, J., Kiernan, K., Yuan, C. J. and Cantara, L.** (2000). The Digital Atheneum: New Techniques for Restoring and Preserving Old Documents. *Computers in Libraries,* **20**(2) (February): 26–31.

**Stamatopoulos, N., Louloudis, G. and Gatos, B.** (2010). Efficient Transcript Mapping to Ease the Creation of Document Image Segmentation Ground Truth with Text-Image Alignment. In *Proceedings of 12th International Conference on Frontiers in Handwriting Recognition,* Kolkata, 16–18 November 2010. IEEE, pp. 226–31.

**Ukkonen, Esko.** 1985. Algorithms for Approximate String Matching. *In­formation and Control,* **64**: 100–118.

**Van Hulle, D., Nixon, M. and Neyt, V.** (2011). Samuel Beckett Digital Manuscript Project. http://www.beckettarchive.org/demo/.