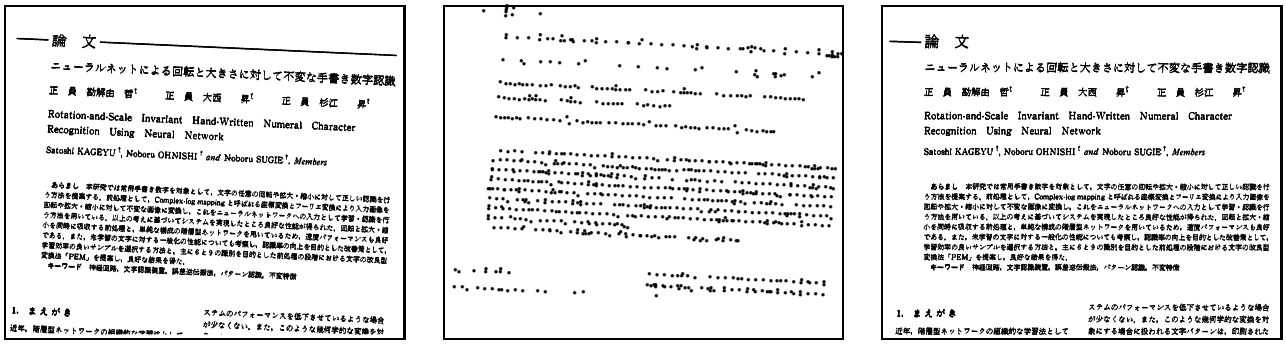

Figure 1: Left - original Japanese page; skew angle is $-2.5°$. Center - fiducial marks for black components; large components are ignored for skew correction. Right - page after pseudo-rotation (ignored components not shown).

Fourier energy spectrum of the page image. Using Parseval's Lemma, this measure can be computed cheaply by projections; also, instead of projecting black pixels, we use the set of centers of the black 8-connected components. An iterative optimization procedure locates the angle of maximum alignment without having to compute the measure over the full range of angles. The resulting estimate of skew is almost always accurate to within ± 3 minutes of arc. It is sensitive enough to detect the presence of a few justified margins of text blocks.

We may not know the dominant orientation, and thus we conduct two searches, one within $10°$ of horizontal, the other $\pm 10°$ of vertical. The angle producing the globally maximum alignment is used to pseudorotate the image. Pseudorotation first rotates the set of centers of the bounding boxes of connected components, and then translates each component to its new position, leaving the bitmaps unrotated. This is fast and avoids aliasing distortions that can result from rotating bilevel images by small angles.

Having corrected the page skew by rotation, we then verify that the alignment angle in the other orientation is close to orthogonal: if not, a second correction is made, by pseudoshearing in a similar way. The result is an image that is skew and shear corrected to a small fraction of a degree. Figure 1 illustrates the technique on a Japanese journal page. Locating 8-connected components (for the entire page) required 5.6 CPU s; skew estimation and correction required an additional 2.2 CPU s.

Later, the procedure is repeated for each block of text. This is not redundant, since even in professionally printed material blocks of text are sometimes pasted up by hand.

Several aspects of this procedure have proved to be particularly important for reliable processing of multilingual documents. Using the magnitude of the alignment measure function to control the sequence of corrections helps ensure that the most important orientation enjoys the most accurate correction. Using the centers of component's bounding boxes as fiducials during skew and shear correction avoids misalignments that can have serious consequences later. Perhaps most importantly, the procedure behaves identically when the page is rotated by any multiple of $90°$. The method has has proven so reliable, in experiments on thousands of pages, that we have come to take it for granted: it fails on perhaps one page in a thousand.

4 Isolating Blocks in a Page

We segment pages, after skew- and shear-correction, into blocks of text — a process sometimes called *zoning* — using a method of greedy white covers ([3], [5]). Our approach is motivated by two observations. The first is that *white space is used as a layout delimiter* in similar ways by publishers and printers in many languages, due to nearly universal conventions of legibility and constraints of printing technology [6]. The second is

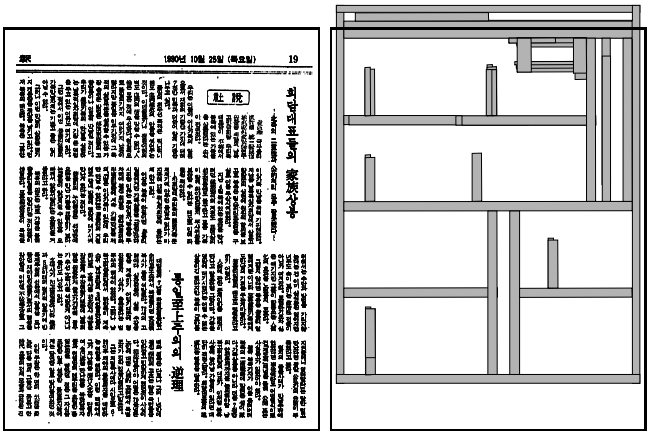

Figure 2: Left: this Korean layout has most, but not all, of the text in vertical text lines. Right: white blocks represent segmentations; blocks of both orientations are correctly segmented.

that *background is simpler than foreground* in many document images: that is, the structure of white space is easier to analyze automatically than local relations among black components.

Our method locates large elongated white rectangles and pieces them together to separate the blocks of text. First, we temporarily set aside components much too big or small to be symbols. Then, all maximal white rectangles – *white covers* – lying between the bounding boxes of the remaining black components are explicitly enumerated. Finally the white covers are sorted in a special order, and unified one by one greedily until a stopping rule is satisfied. The result is a partition of the layout into covered and uncovered areas. Disconnected regions in the uncovered area are identified as text blocks.

By design, the algorithms and heuristics are oblivious to page, block, or text-line orientation: they behave identically when the layout is rotated by multiples of $90°$, or even mirror-imaged.

In trials on 100 English journal layouts representing thirteen publishers and at least 22 distinct styles, 94% of the layouts were segmented correctly. We have also run trials on an unsystematic collection of layouts printed in non-Latin writing systems, including some with vertically-oriented text lines. Since our method is oblivious of orientation, it tends to work equally well on these. Figure 2 shows an Korean-language example of this.

It is important to note that the method works without requiring any prior detailed layout model, either geometric or functional. Runtime on the English-language journal pages averaged 1.8 CPU s, with a maximum of 3.1 s.