Text Line Segmentation using Anisotropic Gaussian Filtering and Watershed

Berat Kurar-Barakat, Sharva Gogawale, Mohammad Suliman, and Nachum Dershowitz

Tel Aviv University

1 Methodology

1.1 Background

Our methodology relies on using the anisotropic Gaussian filter [1]. A *Gaussian filter* smooths an image by convolving it with a bell-shaped kernel whose standard deviation σ is identical in all directions. In contrast, an *anisotropic* Gaussian allows for two distinct standard deviations, σ_u and σ_v, along orthogonal axes (u, v) with the kernel rotated by an angle φ. This produces an elongated Gaussian that smooths more along one axis while preserving structures orthogonal to it.

Taking spatial derivatives of the anisotropic Gaussian prior to convolution transforms the smoother into a feature detector: first-order derivatives emphasize edge transitions, whereas second-order derivatives emphasize ridge and valley like structures (bright-dark-bright or dark-bright-dark patterns). Because differentiation commutes with convolution, the effect is equivalent to differentiating the image after it has been scale-selectively smoothed. This yields noise-robust, orientation-specific responses [1]. By adjusting the derivative order, the orientation φ, and the scale pair σ_u, σ_v, one can detect elongated features such as text lines, vessel-like patterns, or ridges in fingerprint images.

1.2 Pipeline

Our method segments text lines in historical document images using a two-stage pipeline (Fig. 1). The first stage detects the spatial locations of text lines, and the second stage segments the actual text line pixels using these detected line structures as guidance.

The input grayscale image is first binarized using Otsu's thresholding. To enhance line continuity, morphological dilation is applied to the binary image.

We estimate the average character height range by analyzing the bounding boxes of connected components that fall within a plausible vertical size interval. Anisotropic Gaussian filtering is applied with scale parameters derived from the average character height and a fixed elongation factor ($\eta = 3$). This enhances elongated, line-like structures while suppressing non-linear or noise-like components.

To handle merged lines across adjacent text columns, we compute a vertical distance map by summing the distances from each white pixel to the nearest ink pixels above and below. This map is thresholded to generate a vertical separator mask, which is used to eliminate merging line blobs across columns.

The resulting line blobs are vertically dilated using a structuring element proportional to the estimated character height and intersected with the inverted binary mask to isolate text regions from noise and non-text areas. These refined line blobs are then used as markers to guide a watershed algorithm, which segments the pixels of individual text lines from the detected text regions.

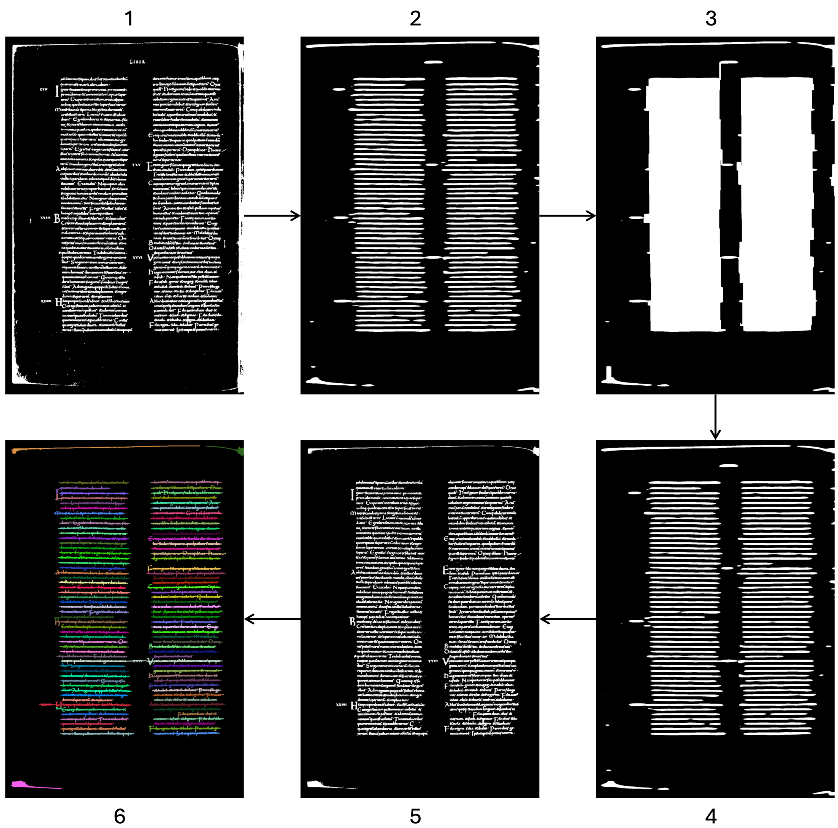

Fig. 1. Text line segmentation pipeline. A grayscale input image is first binarized and dilated (1). Anisotropic Gaussian filtering, applied at scales corresponding to the average character height, detects the text lines (2). A vertical separator mask (3) removes merged text line detections across columns (4). The resulting blobs are vertically elongated to detect text regions (5). Finally, the detected lines serve as markers to guide a watershed algorithm, segmenting individual text lines from the detected text regions (6).

Table 1. Performance of the proposed method on Latin 2, Latin 14396, Syriaque 341, and their average across all evaluation metrics.

Metric	**Latin 2**	**Latin 14396**	**Syriaque 341**	**Average**
Pixel IU	0.674	0.739	0.649	0.687
Line IU	0.792	0.946	0.594	0.777
Detection rate (DR)	0.598	0.609	0.326	0.511
Recognition accuracy (RA)	0.636	0.645	0.319	0.533
F-measure (FM)	0.613	0.626	0.315	0.518

2 Results

Table 1 presents the performance evaluation of our method on the text line segmentation task. Results are reported per manuscript as well as averaged across three manuscript classes: Latin 2, Latin 14396, and Syriaque 341. The evaluation is based on the metrics specified in the competition: pixel intersection over union (Pixel IU); line intersection over union (Line IU), detection rate (DR); recognition accuracy (RA); and F-measure (FM).

3 Conclusion

The proposed method yields an average Pixel IU of 0.687 and a Line IU of 0.777 across the three evaluated manuscripts. Performance values are lower on the Syriaque 341 manuscript across multiple metrics, including Line IU, detection rate, recognition accuracy, and F-measure, which reflects the increased difficulty posed by its layout complexity and degradation. In contrast, Latin 14396 yields the highest metric values among the three datasets.

Acknowledgments. This research was funded in part by the European Union (ERC, MiDRASH, Project No. 101071829, principal investigators: Daniel Stökl Ben Ezra, EPHE-PSL; Judith Olszowy-Schlanger, EPHE-PSL; Nachum Dershowitz, Tel Aviv University; Avi Shmidman, Bar-Ilan University). Views and opinions expressed are, however, those of the authors only and do not necessarily reflect those of the European Union or the European Research Council Executive Agency. Neither the European Union nor the granting authority can be held responsible for them.

References

1. Geusebroek, J.M., Smeulders, A.W.M., van de Weijer, J.: Fast anisotropic Gauss filtering. IEEE Transactions on Image Processing **12**(8), 938–943 (Aug.). https://doi.org/10.1109/TIP.2003.812429