


Article

Boundary Gaussian Distance Loss Function for Enhancing Character Extraction from High-Resolution Scans of Ancient Metal-Type Printed Books

Woo-Seok Lee  and Kang-Sun Choi * 

Future Convergence Engineering, Department of Electrical Electronics and Communication Engineering, Korea University of Technology and Education, Cheonan 31253, Republic of Korea; book0496@koreatech.ac.kr

* Correspondence: ks.choi@koreatech.ac.kr

Abstract: This paper introduces a novel loss function, the boundary Gaussian distance loss, designed to enhance character segmentation in high-resolution scans of old metal-type printed documents. Despite various printing defects caused by low-quality printing technology in the 14th and 15th centuries, the proposed loss function allows the segmentation network to accurately extract character strokes that can be attributed to the typeface of the movable metal type used for printing. Our method calculates deviation between the boundary of predicted character strokes and the counterpart of the ground-truth strokes. Diverging from traditional Euclidean distance metrics, our approach determines the deviation indirectly utilizing boundary pixel-value difference over a Gaussian-smoothed version of the stroke boundary. This approach helps extract characters with smooth boundaries efficiently. Through experiments, it is confirmed that the proposed method not only smoothens stroke boundaries in character extraction, but also effectively eliminates noise and outliers, significantly improving the clarity and accuracy of the segmentation process.

Keywords: loss function; boundary distance; denoising; character segmentation; historical documents



Citation: Lee, W.-S.; Choi, K.-S. Boundary Gaussian Distance Loss Function for Enhancing Character Extraction from High-Resolution Scans of Ancient Metal-Type Printed Books. *Electronics* **2024**, *13*, 1957. <https://doi.org/10.3390/electronics13101957>

Academic Editor: Xenophon Zabulis

Received: 4 April 2024

Revised: 11 May 2024

Accepted: 15 May 2024

Published: 16 May 2024



Copyright: © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Old printed documents play a crucial role in understanding historical perspectives through textual analysis, serving as windows to historical events. They provide insights into the social and cultural facets of history, as well as ancient civilizations [1]. Recently, researchers have increasingly focused on knowledge extraction by analyzing ancient printed documents. The archaeological knowledge extraction includes not only analysis of text content, but also evaluation of metal-type production technology through face estimation and comparison of metal type, as well as printing technology assessment through printing quality inspection.

However, due to their age, these printed documents often suffer from information loss [2]. Their poor preservation, coupled with their antiquity, has adversely affected the contained information [3], presenting a significant challenge for researchers attempting to extract accurate information. To address this, technological advances in machine learning and computer vision algorithms are being employed, particularly in character enhancement and segmentation techniques [4].

Despite the inception of image processing for manuscript retrieval several decades ago, a fully reliable system for accurate knowledge extraction from these documents is still elusive [5]. For successful knowledge extraction, accurate character segmentation is essential. It is notable that character recognition generally follows character segmentation, and, thus, character recognition accuracy is an important measure rather than the accuracy of the character shape itself [2]. In this case, because it is enough to be able to read character shape, the quality of the segmented characters is not necessarily high, and, thus, the scan resolution is not very high.

Jeong et al. developed a system for analyzing old metal-type printed documents for reconstructing 3D movable metal types that were used in printed materials, but it has been lost [1]. Unlike conventional methods, character segmentation herein, according to the purpose of the system, aims to determine the exact character shape in order to accurately estimate the typeface of metal type from high-resolution scans of old metal-type printed documents. Various challenges need to be addressed in this scenario.

Figure 1 shows two high-resolution scanned pages of a historic 15th-century metal-type printed book “Geun-sa-rok vol. 6”. By folding a larger printed piece of paper, the two pages end up on the reverse side of each other. The images exhibit not only the content but also unique features, such as the ‘outer line’, ‘separating line’, ‘central part of the typesetting board’, and ‘collection seal’ [6]. The separating lines are the imprints of long bamboo strips inserted between horizontally adjacent metal types to secure the types. The central part of the board indicates the book title and page number. These components, extraneous to the primary content, should be removed through the character segmentation process.

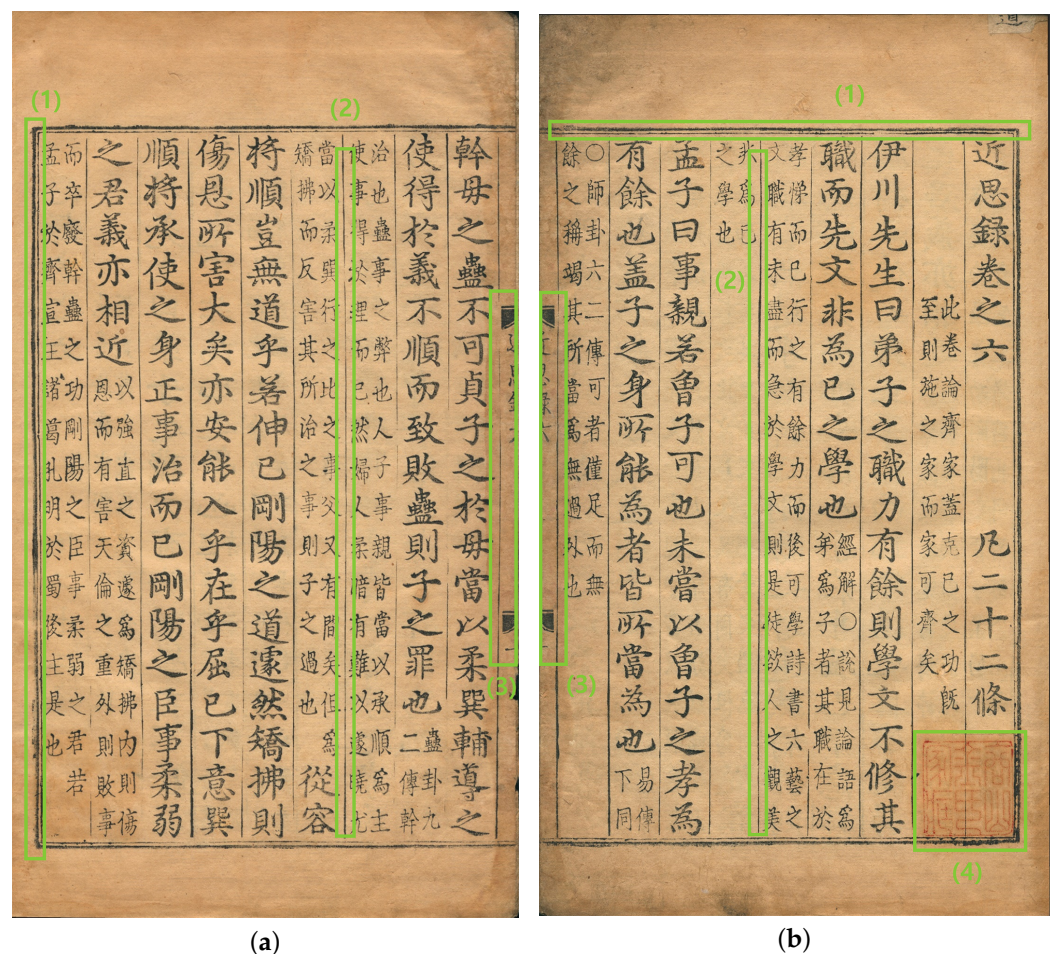


Figure 1. Examples of scanned pages in Korea’s 15th-century book “Geun-sa-rok vol. 6”, printed using movable metal types. (a) The second page of the book. (b) Its first page, which is the reverse side of (a). The content of the two pages is actually printed on a single large piece of paper. After printing, the paper is folded in half into two pages. The printed page contains not only the text but also several components including (1) the outer line of the typesetting board, (2) (column) separating lines, (3) the central part of the board, and (4) a collection seal. For textual analysis, character segmentation should extract only the text, excluding other components.

Another challenge in extracting characters from the documents involves dealing with inconsistency in print quality, as depicted in Figure 2a. The unique Korean printing method,

differing from Gutenberg’s European pressing technique, involved hand-pressing the paper with wood sticks or cotton lumps, often resulting in unclear lettering.

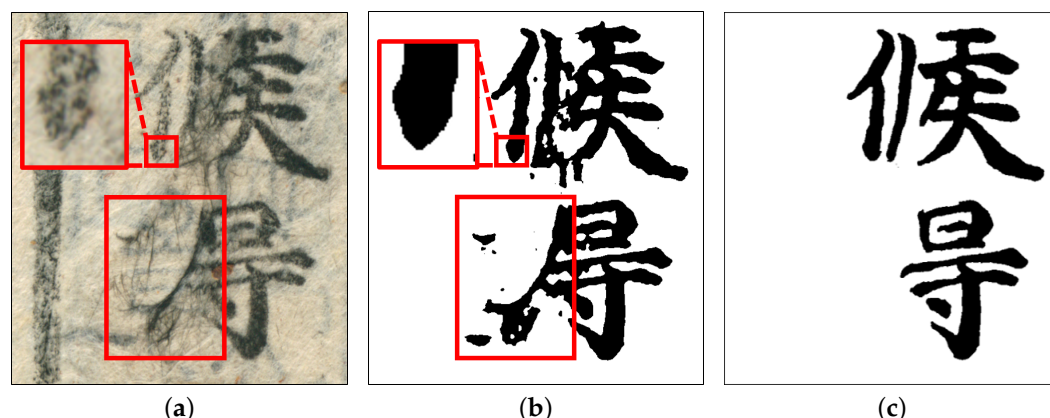


Figure 2. Example of a low-quality printed document image. (a) Scanned image. (b) Character extraction result obtained using a conventional segmentation network. (c) Ground-truth image.

Uneven ink coverage on the face of metal types results in noisy ‘dots’ and ‘holes’, as shown in Figure 2a. The upper enlarged red box shows unlinked holes within a character stroke. Inconsistent hand pressure causes insufficient inking or slight movement of types, leading to blurred prints. Ink bleed can occur along long-fiber pulps of the paper as in the lower box. In that case, character segmentation networks trained with a conventional loss function easily produce unsatisfactory results of rough strokes, noisy dots, and holes, as illustrated in Figure 2b.

In addition, the aforementioned low print quality risks inconsistent interpretation in producing reliable GT labels. When building a new dataset, it is widely used to modify the results obtained using pretrained models, due to the high cost of labeling high-resolution images. That is, the images are initially masked using a pretrained segmentation model like Figure 2b, and then the masks are corrected by trained workers. Although large errors in the initial masks will be corrected, minor details may not be handled properly. In particular, the ambiguity of the stroke boundaries leads to rough and bumpy boundaries in the pretrained model predictions, which persist even after correction, as shown in Figure 2c.

This paper proposes a novel loss function, the boundary Gaussian distance (BGD) loss, designed to enhance character extraction accuracy tailored for ancient Korean movable metal-type printed documents. Specifically, our aim is to estimate the face shape of movable metal type used to print each character in documents.

The BGD loss estimates deviation between the boundary of predicted character strokes and the counterpart of GT. The proposed loss function determines the deviation by indirectly exploiting boundary pixel-value difference over a Gaussian-smoothed version of the GT boundary for computational efficiency, instead of the time-consuming method of searching for corresponding pixels between two boundaries and calculating distance of the correspondences. As mentioned earlier, when producing GT for high-resolution document images, the stroke boundaries in the GT can be rough and bumpy. The usage of the smoothed version of the GT also alleviates the roughness of stroke boundaries and, thus, helps the segmentation model produce strokes with smooth boundaries. This study underscores the effectiveness of the Gaussian distance loss technique in denoising, thereby facilitating more efficient information retrieval from ancient texts [7].

The rest of this paper is presented as follows. In Section 2, we delve into related work to review character segmentation methods and various loss functions. In Section 3, the proposed loss function is described in detail and the importance of this technique is emphasized, highlighting the significance of the BGD loss approach. In Section 4, the effectiveness of the proposed method is demonstrated through various experiments. Finally, we conclude this paper.

2. Related Work

2.1. Character Segmentation

The analysis of the scholarly articles showed that character segmentation supports the evaluation of ancient documents especially those that have lost a major part of information. The research includes three main stages in the character segmentation that are needed to obtain the desired outcome: text-line segmentation, word segmentation, and character segmentation [8]. For handwritten scripts, letter sizes can vary greatly and multiple letters are connected together. Therefore, line segmentation and word segmentation are important because precise line and word splitting helps with accurate character segmentation and the following character recognition. The majority of the scholarly articles discussed the significance of character segmentation by using the Arabic, Thai, Chinese, and Japanese scripts, but only a few of them worked on the Korean language. Ancient Korean documents are rarely investigated, in comparison with those in the modern Korean language, because of the complexity of character formation and difficulty in segmentation [9]. Kim et al. studied Chinese character segmentation by focusing mainly on the recognition-based method for character segmentation of handwritten text [10]. For the sake of reducing recognition errors, the researchers focused on using the characters' geometric features and context information. It is noteworthy that most character segmentation in this application aims to determine a bounding box for each character.

However, for documents printed using movable metal type, such text-line division, and even character segmentation, become easy tasks because the size of the movable metal type used is similar and each letter becomes disconnected distinctly. Jeong et al. developed a system for analyzing old metal-type printed documents for reconstructing 3D models of movable metal types that were used in printed materials, but their model has been lost [1]. By precisely comparing shapes of characters that have been segmented, researchers can estimate how many individual metal types were used to print a book, which allows them to evaluate the level of metal-type printing technology at the time.

Recent researches on character segmentation involves natural-scene text segmentation, including text-region extraction, character recognition, and features of characters font [2,11–19]. Tang et al. proposed a three-stage CNN-based model, in which candidate text regions were detected, refined, and filtered in those stages correspondingly [20]. SMANet, by adopting the encoder–decoder structure of PSPNet [21], created a new multiscale attention module for accurate text segmentation [22]. A segmentation network, TexRNet, was jointly proposed with a dataset for natural-scene text segmentation [23]. Trimap and glyph discriminator losses tackled diverse texture and arbitrary scales/shapes, leading to improvement. Scene text segmentation focuses on word recognition in a manner that is robust to geometric distortions and character variations in size, color, and font, whilst our pixel-wise character segmentation focuses on accurately extracting the shape of character strokes in order to compare the typefaces of the metal type used.

2.2. Loss Functions

In the field of image segmentation, various loss functions have been developed to measure how different the segmentation result \hat{Y} is from the GT label Y . Cross-entropy (CE) is derived from Kullback–Leibler divergence, which is a measure of dissimilarity between two distributions [24]. The CE loss uses this metric as a loss function and is predominantly utilized in classification tasks, functioning to categorize the type of objects. It compares the probability distributions of the network's prediction \hat{Y} with Y . Equation (1) formulates the CE loss, where n represents the total number of pixels in the image and $P(\hat{Y})$ denotes the probability of \hat{Y} :

$$L_{CE}(\hat{Y}, Y) = -\frac{1}{n} \sum Y \log P(\hat{Y}). \quad (1)$$

The Dice loss, which is widely employed, yields lower values as the overlap between the regions of Y and \hat{Y} increases [25]. Equation (2) represents the Dice score (DSC), a region

similarity measure, that utilizes the extent of overlap between areas \hat{Y} and Y . The DSC escalates as the intersection of \hat{Y} and Y enlarges. Equation (3) represents the loss function based on the DSC.

$$S_D(\hat{Y}, Y) = \frac{2|\hat{Y} \cap Y|}{|\hat{Y}| + |Y|}. \tag{2}$$

$$L_D(\hat{Y}, Y) = 1 - S_D(\hat{Y}, Y) = 1 - \frac{2|\hat{Y} \cap Y|}{|\hat{Y}| + |Y|}. \tag{3}$$

Mathieu et al. worked on pixel-spaced video prediction by adopting an unsupervised machine learning method and defining an image gradient-difference loss function [26]. Different from total variation loss [27], the gradient-difference (GD) loss considers the difference between the gradient magnitude of the prediction image \hat{Y} and GT image Y for denoising [26].

The GD loss is defined as

$$L_{GD}(\hat{Y}, Y) = \sum_{i,j} \left(\left| |\hat{Y}_{i,j} - \hat{Y}_{i+1,j}| - |Y_{i,j} - Y_{i+1,j}| \right|^\alpha + \left| |\hat{Y}_{i,j} - \hat{Y}_{i,j+1}| - |Y_{i,j} - Y_{i,j+1}| \right|^\alpha \right), \tag{4}$$

where α is a hyperparameter value that is greater than 1 and (i, j) represents the pixel position in the image.

The boundary gradient-consistency (BGC) loss is designed to ensure the consistency of character boundary gradient vectors [28]. The BGC loss function was developed specifically for smoothing the boundary lines of characters printed using the Korean movable metal type. To do this, the BGC loss penalizes rapid changes in the gradients along character stroke boundaries by comparing the gradient direction of an anchor (center) pixel with the maximum gradient magnitude among its neighboring pixels. The BGC loss is defined as

$$L_{BGC}(\hat{Y}) = \frac{1}{|\hat{\mathcal{V}}|} \sum_{(i,j) \in \hat{\mathcal{V}}} \left(\left| \nabla \hat{Y}_{i,j} - \nabla \hat{Y}_{i^*,j^*} \right| \right), \tag{5}$$

where $\hat{\mathcal{V}}$ is the set of boundary pixel positions in the prediction \hat{Y} . (i^*, j^*) indicates the position of the locally maximum gradient magnitude, which is defined as

$$(i^*, j^*) = \arg \max_{(k,l) \in \mathcal{N}(i,j)} \left| \nabla \hat{Y}_{k,l} \right|, \tag{6}$$

where $\mathcal{N}(i, j)$ is a set of neighboring pixels around anchor pixel (i, j) and $\left| \nabla \hat{Y}_{k,l} \right|$ is the gradient magnitude at (k, l) .

The Hausdorff distance measures the extent to which each point of a set lies near some point of the other set and vice versa [29]. Therefore, this distance can be used to determine the degree of resemblance between two objects. The Hausdorff loss uses the Hausdorff distance itself as a loss function, and is defined as

$$L_H(Y, \hat{Y}) = \max \left(\tilde{\delta}_H(Y, \hat{Y}), \tilde{\delta}_H(\hat{Y}, Y) \right), \tag{7}$$

where the set of minimal distances is defined as

$$\tilde{\delta}_H(Y, \hat{Y}) = \max_{y \in Y} \min_{\hat{y} \in \hat{Y}} \|y - \hat{y}\|. \tag{8}$$

This method involves, for every boundary pixel in the GT, finding the closest predicted boundary pixel and then identifying the maximum value within the set of these shortest distances. The process is then repeated with the predicted boundary as the reference. Finally, the larger of the two maximum values is used as the final loss value. If the set of

minimal distances contains outliers, it may skew the Hausdorff distance, as it does not accommodate any other distance due to the outlier, potentially failing to accurately measure the distance between the two images. The average Hausdorff distance is proposed as a solution to mitigate the influence of outliers on the results when computing the Hausdorff distance. However, this approach requires a substantial computational cost because it determines the distances between pixels using Euclidean distance in a brute-force manner.

To summarize the loss functions mentioned earlier, the Dice and the CE loss functions are regarded as *region-based loss* in that all the pixels of image are utilized for calculation. In contrast, the GD, the BGC, and the Hausdorff loss functions are *boundary-based losses* because they only focus on boundary pixels and measure deviation between the boundaries, as shown in Figure 3.



Figure 3. Boundary-based distance between GT and prediction. (a) A GT Y and a prediction \hat{Y} . (b) Overlay of Y and \hat{Y} . Although the GT and prediction are actually binary images, for easy understanding, Y and \hat{Y} are represented in orange and green, respectively. Boundary-based distance focuses on how far apart the pixels of two boundaries are.

When the GD loss is used for natural images, all the pixels can be used, as gradient values would be non-zero. However, using binary images, as shown in Figure 2, actually results in only non-zero gradients along object boundaries. Therefore, the GD loss is regarded as boundary-based loss in this paper.

3. Proposed Boundary Gaussian Distance Loss

The BGD loss is proposed to segment characters in old metal-type printed documents. In particular, the proposed loss helps the segmentation network generate character strokes with smooth boundaries in the presence of inaccurate GT labels. Considering rough and bumpy boundaries of GT labels, the loss associated with accurately measuring geometric distance between the boundaries of GT and prediction may cause the network to segment characters with bumpy boundaries. In order to alleviate this problem, difference between the boundaries of GT and prediction is measured after smoothing in the proposed boundary distance-based loss.

Figure 4 shows the process performed in the proposed loss function. Initially, for two image segments Y and \hat{Y} to be compared, their boundary images B and \hat{B} are obtained using simple morphological operations as follows:

$$B = Y - E(Y), \quad (9)$$

$$\hat{B} = \hat{Y} - E(\hat{Y}), \quad (10)$$

where $E(\cdot)$ is the morphological erosion operation.

Although Y and \hat{Y} are actually binary images, they are colored for easy understanding. For notational clarity, we denote boundary pixel positions by $\mathcal{V} = \{(x, y) | B(x, y) > 0\}$.

For B and \hat{B} , smoothed fields G and \hat{G} are respectively yielded by performing Gaussian smoothing

$$G = B * \mathbb{G}_\sigma, \quad (11)$$

where $*$ is the convolution operation and G_σ represents the Gaussian function, given as

$$G_\sigma(x, y) = \exp\left(-\frac{x^2 + y^2}{2\sigma^2}\right). \tag{12}$$

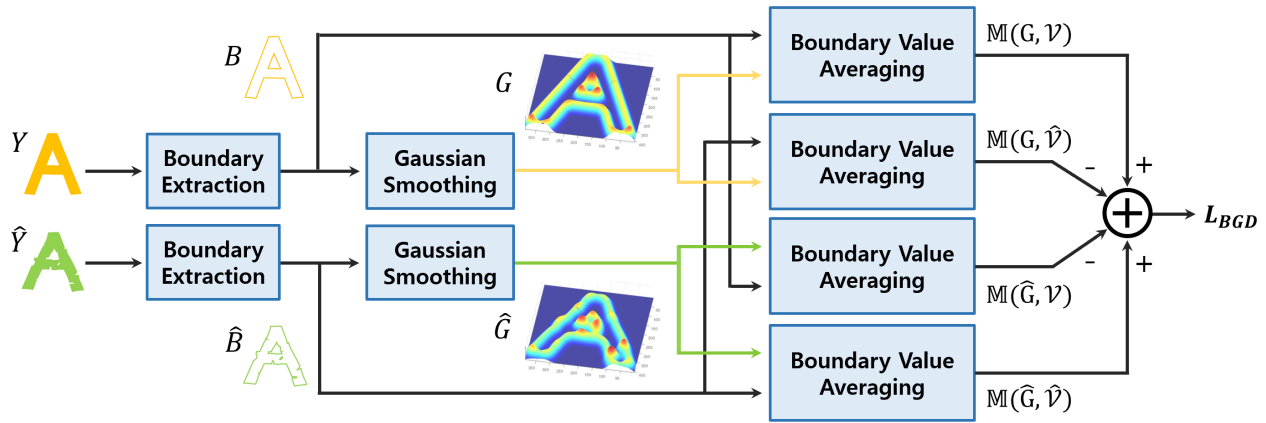


Figure 4. The process of the proposed boundary Gaussian distance loss function. For easy understanding, two input binary images are colored.

In the proposed method, the distance between two boundaries is approximately estimated based on the smoothed field. Figure 5 illustrates how the distance is calculated using the field. Figures 5a,b show two boundaries to be compared, \mathcal{V} and $\hat{\mathcal{V}}$, and G , respectively. As G is the smoothed version of B , the value of G would be locally highest at the pixel location on \mathcal{V} . The farther a pixel location is from \mathcal{V} , the more its value on G decreases. Figure 5c plots G values along the red dashed horizontal line and boundary-pixel positions corresponding to the line to demonstrate this phenomenon. In the parts of ‘A’ and ‘B’ in Figure 5d, corresponding pixels are distant, and, thus, the difference in pixel values become large, while the correspondences within ‘C’ and ‘D’ are close to each other, leading to small pixel-value difference. Based on this observation, in the proposed method, the amount of pixel-value difference along the boundaries of two shapes is utilized as a measure of subtle shape difference.

Defining an average of the pixel values of G along the boundary \mathcal{V} as

$$\mathbb{M}(G, \mathcal{V}) = \frac{1}{|\mathcal{V}|} \sum_{(i,j) \in \mathcal{V}} G(i, j), \tag{13}$$

an average pixel-value difference between \mathcal{V} and $\hat{\mathcal{V}}$ can be obtained by $\mathbb{M}(G, \mathcal{V}) - \mathbb{M}(G, \hat{\mathcal{V}})$. Here, $|\cdot|$ represents the cardinal of a set.

The proposed BGD loss function consists of double calculation of the average pixel-value difference. Specifically, one is obtained over G and the other is calculated over \hat{G} as

$$L_{BGD} = [\mathbb{M}(G, \mathcal{V}) - \mathbb{M}(G, \hat{\mathcal{V}})] + [\mathbb{M}(\hat{G}, \hat{\mathcal{V}}) - \mathbb{M}(\hat{G}, \mathcal{V})], \tag{14}$$

where $\mathbb{M}(G, \mathcal{V}) \geq \mathbb{M}(G, \hat{\mathcal{V}})$ and $\mathbb{M}(\hat{G}, \hat{\mathcal{V}}) \geq \mathbb{M}(\hat{G}, \mathcal{V})$.

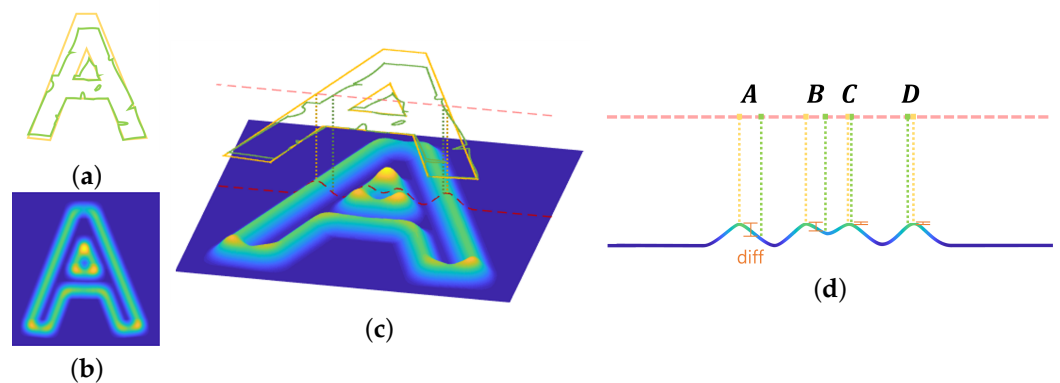


Figure 5. Boundary distance approximation using smoothed boundary field G . (a) Two boundaries to be compared, \mathcal{V} and $\hat{\mathcal{V}}$. (b) Smoothed field G . (c) \mathcal{V} and $\hat{\mathcal{V}}$ overlaid over G . (d) Boundary-pixel positions over G in a horizontal line. The distance between the two boundary pixels is approximated by the difference of G values at the pixels. When two boundary pixels are far apart, as in ‘A’ and ‘B’, the difference of G values increases. When two boundaries are close to each other, as in ‘C’ and ‘D’, the difference becomes negligible.

Although both the average pixel-value differences indicate boundary distance as well, each contributes differently within the loss function. Figure 6 shows various wrong prediction cases. In Figure 6a, where the prediction is estimated to be slightly thinner or thicker than GT, the lengths of the two boundaries are almost the same ($|\mathcal{V}| \approx |\hat{\mathcal{V}}|$), but pixel-value difference occurs and, thus, $\mathbb{M}(G, \hat{\mathcal{V}})$ and $\mathbb{M}(\hat{G}, \mathcal{V})$ become smaller than $\mathbb{M}(G, \mathcal{V})$ and $\mathbb{M}(\hat{G}, \hat{\mathcal{V}})$, respectively. Consequently, both the average pixel-value differences in the loss function increase similarly. If a hole occurs falsely inside a stroke, as shown in Figure 6b, the second average pixel-value difference is almost zero (because $\mathbb{M}(\hat{G}, \hat{\mathcal{V}}) \approx \mathbb{M}(\hat{G}, \mathcal{V})$). Since, however, the hole increases the number of boundary pixels whose values are zero on G , $\mathbb{M}(G, \hat{\mathcal{V}})$ shrinks, and the consequential increment in the first average pixel-value difference leads to an increase in the loss value. If a dotted noise is falsely detected, the first average pixel-value difference increases. Contrarily, if a stroke is totally missed, as shown in Figure 6d, the second average pixel-value difference increases, while the first becomes almost zero.

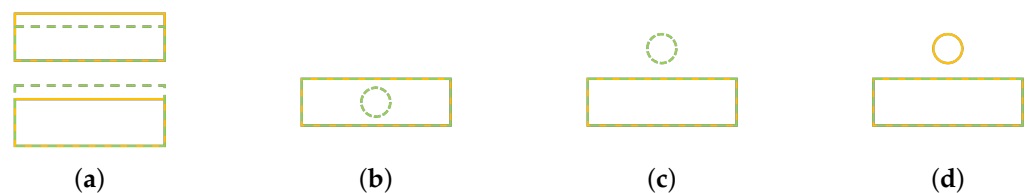


Figure 6. Wrong prediction cases. GT and prediction strokes are indicated by yellow and green dashed lines, respectively. (a) Prediction is slightly thinner or thicker than GT. (b) A hole appears inside a predicted stroke. (c) A noisy stroke of a green circle is falsely detected. (d) A circular stroke is falsely missed.

It is notable that $\mathbb{M}(\cdot, \cdot)$ is bounded well because of the normalization with $|\mathcal{V}|$ in Equation (13). Without the normalization, pixel values accumulate significantly as the number of boundary pixels increases. Such boundedness of $\mathbb{M}(\cdot, \cdot)$ allows the proposed loss function to be utilized with other loss functions. The proposed approach is computationally efficient in that it does not require the determination of the nearest location to measure distance.

In the proposed boundary distance approximation, the smoothed boundary field is obtained using the Gaussian filter. The benefits of the Gaussian filtering are twofold: (1) As

mentioned earlier, the roughness of stroke boundaries in the GT is alleviated by smoothing, and, thus, it leads the segmentation network to produce smooth stroke boundaries; (2) Due to the properties of Gaussian functions, the proposed loss function is effective for the type of segmentation errors encountered in our application. Figure 7 plots the first order partial derivative of the 2-D Gaussian filter:

$$\frac{\partial \mathbb{G}_\sigma(x, y)}{\partial x} = -\frac{x}{\sigma^2} \mathbb{G}_\sigma(x, y) \quad (15)$$

and its intersection at $y = 0$. Assume that the boundary pixel of the GT stroke is located at $x = 0$. Considering that the derivative represents the amount of change in the function, when the predicted boundary pixel is located near zero, the increment in the proposed loss function is small. However, the loss significantly increases near $x = \sigma$ since the derivatives of the Gaussian function have the extreme values at $x = \sigma$. That is, small deviations of the predicted boundary from GT are minimally penalized, while deviations of σ are severely penalized by the proposed loss function. This shows that the proposed loss is well suited to the types of segmentation errors that occur when extracting characters from old metal-type printed documents.

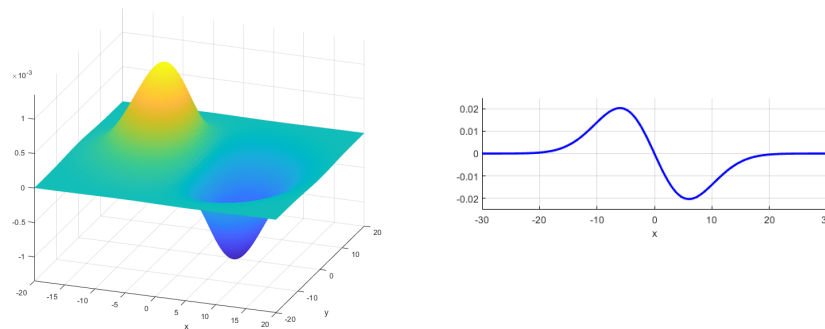


Figure 7. The 1st-order derivative of the 2-D Gaussian filter of $\sigma = 6$ and its intersection at $y = 0$.

4. Experimental Results

4.1. Dataset

We scanned the metal-type printed documents that are summarized in Table 1. To minimize scanning distortion, a flatbed scanner was used. Figure 1 depicts an example of a scanned image whose size is about 6000×8000 pixels. For building a high-resolution image dataset of characters printed with old movable metal types, we selected 10 scanned page images from each of the last three books in Table 1 and performed the segmentation labeling. As mentioned earlier, the scanned page images were initially masked using a pretrained segmentation model, and then the masks were corrected by trained workers.

Table 1. Books scanned for building a high-resolution image dataset.

Book Title	Movable Metal Type Used	Sheets	Published Year	Resolution
Jikji	Heung-deok-sa-ja	39	1377	600 dpi
Seokbo-sangjeol vol. 6	Kab-in-ja	47	1447	600 dpi
Worin-cheongang-jigok	Kab-in-ja	71	1447	600 dpi
Suneung-eomgyeong (eonhae) vol. 4	Eul-hae-ja	115	1462	600 dpi

We then randomly cropped the page image and mask label with random rotation and scale augmentation to generate 13,500 pairs of image patches and their corresponding labels, with a size of 1024×1024 pixels. The pairs were separated into a training set of 12,150 pairs and a test set of 1350 pairs. For better generalization, random crop was

performed once again when loading the pairs. Specifically, the actual size to be fed into the network is 512×512 within 1024×1024 .

Typically, two sizes of metal type have been used for printing. Because the books were written vertically, one large type or two small types had been placed in one column. The average thickness of character strokes is about 30 pixels for large letter types and 10 pixels for small types.

4.2. Analytic Study

For all experiments, the Trans-UNet architecture was utilized for character segmentation. Each model was trained with a batch size of 38 for 40 epochs using two A6000 GPUs. α for L_{GD} was set to 1 in our experiments. As explained in the next subsection, the proposed BGD loss function was utilized by combining the Dice loss for the following experiments.

For the Gaussian smoothing in the proposed BGD loss function, σ should be determined. Table 2 summarizes the impact of σ on the segmentation performance in terms of Dice score, S_D , in Equation (2) and the standard deviation of S_D . A lower standard deviation indicates more consistent quality results. The performance change in DSC is very small, but the best results are obtained with σ of 6, which is related to the stroke thickness in the dataset. In σ_{S_D} , the performance is insensitive to σ , but the minimum is obtained when σ is either 6 or 14.

Table 2. Comparison of segmentation performance depending on σ for Gaussian smoothing. The best values are indicated in bold.

σ	DSC $S_D \uparrow$	Standard Deviation $\sigma_{S_D} \downarrow$
2	0.9722	0.0158
6	0.9744	0.0144
8	0.9720	0.0146
10	0.9717	0.0145
14	0.9718	0.0144

The proposed loss function consists of two average pixel-value differences. As described earlier, since the two deal with different problem cases, employing a single average pixel-value difference degrades performance. Table 3 shows the performance of the segmentation models trained using a single average pixel-value difference in terms of the average DSC and σ_{S_D} .

Specifically, $L_{BGD}(G)$ uses only the former $[\mathbb{M}(G, \mathcal{V}) - \mathbb{M}(G, \hat{\mathcal{V}})]$ in Equation (14), whilst $L_{BGD}(\hat{G})$ uses only the latter average pixel-value difference by using \hat{G} , that is, $L_{BGD} = L_{BGD}(G) + L_{BGD}(\hat{G})$. As expected, exploiting both average pixel-value differences successfully improves the segmentation performance in DSC and σ_{S_D} . It is notable that σ_{S_D} of $L_D + L_{BGD}$ is 57% smaller than that of the single usage.

Table 3. Performance comparison of results obtained using different combinations of both the average pixel-value differences within the proposed BGD loss function. The best values are indicated in bold.

Loss Function	DSC $S_D \uparrow$	Standard Deviation $\sigma_{S_D} \downarrow$
$L_D + L_{BGD}(G)$	0.9679	0.0242
$L_D + L_{BGD}(\hat{G})$	0.9677	0.0268
$L_D + L_{BGD}$	0.9744	0.0144

Figure 8 shows the segmentation results when a single average pixel-value difference is used. Both produced holes of different sizes inside the character strokes and the outer line of the typesetting board was not severely removed, leaving a large false-positive region. However, when both were utilized together, as in the proposed function, these problems were successfully overcome.

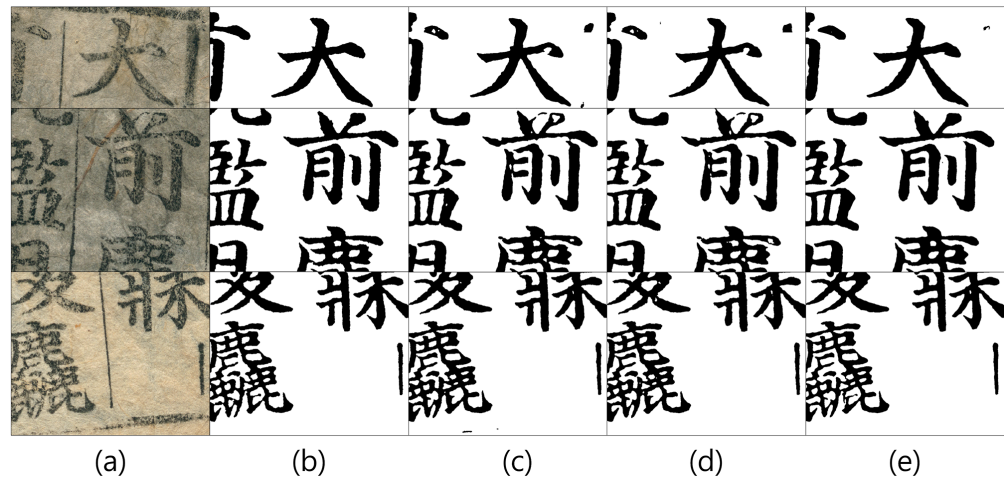


Figure 8. Subjective comparison of results obtained using different combinations of both the average pixel-value differences within the BGD loss function. (a) Scanned images. (b) GT. (c) Result of $L_D + L_{BGD}(G)$. (d) Result of $L_D + L_{BGD}(\hat{G})$. (e) Result of $L_D + L_{BGD}$.

4.3. Objective Comparison

Table 4 summarizes the performance of the segmentation models trained with various combinations of loss functions in terms of the average DSC and σ_{S_D} .

First, we show region-based loss functions that use all the pixels for the loss calculation. The single usage of the Dice loss function showed a high S_D . This is because performance was also evaluated using the same metric. L_D^{10} represents the result of the Dice loss after 10 epochs, which was utilized for training with the boundary-based loss functions. Training for 30 additional epochs improved the performance in S_D and σ_{S_D} . The CE loss exhibited a DSC value similar to L_D , but σ_{S_D} was reduced significantly.

The initial random state of the network leads to randomly noisy initial results. Because the boundary-based loss function focuses around the stroke boundaries of the GT, it is difficult to properly optimize the network with a single use of the boundary-based loss function. To solve this problem, we trained the network using a region-based loss function for the first 10 epochs and then trained the network using the boundary-based loss function of interest for an additional 30 epochs. Such a training strategy for the boundary-based loss functions allowed the network to be trained, but with a slight decrease in DSC compared to L_D^{10} . Similar to the total variation loss widely used for denoising [27], L_{BGC} in Equation (5) only uses prediction and, therefore, cannot be employed alone for loss calculation. Although L_{GD} utilizes GT as well as prediction, when L_{GD} is used for binarized GT and prediction, the optimization fails, unlike when used for image enhancement with natural images, as in [30].

Typically, combinations of loss functions resulted in better performance than their single usage. However, this seems not applicable for this character stroke segmentation. The combination of region-based functions, $L_D + L_{CE}$, even degraded performance a little compared to L_D and L_{CE} .

In the single usage, region-based loss functions achieved higher scores than boundary-based loss functions. However, this tendency was reversed when combining losses. The combination of region-based loss and boundary-based loss optimizes the network jointly in two aspects. When combined with L_D , all boundary-based loss functions significantly improved segmentation performance. Nonetheless, except for $L_D + L_{BGD}$, there still was very slight performance degradation compared to the single usage of region-based functions. In particular, $L_D + L_H$ showed the largest σ_{S_D} , except for the failure cases of L_H and L_{BGD} . However, the proposed loss function achieved excellent performance in both S_D and σ_{S_D} when combined with L_D . This is the only case where performance improved when combining losses.

Compared to the performance increase by combining L_H with L_D , the performance improvement of $L_D + L_{BGD}$ is 57% higher, achieving the highest score in DSC. The combination of the proposed function outperforms the others in σ_{S_D} , yielding the smallest value, which is 82% smaller than the second smallest value of L_{CE} . This means that the model using the proposed function produces consistently successful results for most data.

Table 4. Performance comparison of various loss functions. The best values are indicated in bold.

Characteristics	Loss Function	DSC $S_D \uparrow$	Standard Deviation $\sigma_{S_D} \downarrow$
Single, Regional	L_D^{10}	0.9596	0.0312
	L_D	0.9700	0.0248
	L_{CE}	0.9699	0.0176
Single, Boundary	L_H	0.1473	0.0386
	L_{BGD}	0.1473	0.0386
	$L_D^{10} \rightarrow L_H$	0.9584	0.0319
	$L_D^{10} \rightarrow L_{BGD}$	0.9576	0.0303
Combination, Regional	$L_D + L_{CE}$	0.9642	0.0303
Combination, Boundary	$L_D + L_H$	0.9691	0.0353
	$L_D + L_{GD}$	0.9699	0.0262
	$L_D + L_{BGC}$	0.9693	0.0219
	$L_D + L_{BGD}$	0.9744	0.0144

4.4. Qualitative Comparison

Figure 9 illustrates the segmentation results when different loss functions are used alone for various low-quality prints. Figure 9a shows examples of an identical resolution, while Figures 9b,c enlarge problematic parts and their corresponding GT labels. Overall, the results when using a single loss function are unsatisfactory.

As mentioned earlier, many of the GT's stroke boundaries are not smooth. The first-row image shows rough character boundaries and dots caused by ink splatters. In GT, the ink splatters are removed, but the boundaries are still bumpy. Using a single loss function cannot handle the roughness of the boundary properly. The region-based loss functions detect ink splatters and generate false-positive dots, while the boundary-based loss functions can remove the splatters effectively. In rows 2 and 3, ink bleeding occurs along the fiber-pulp of the paper. The comparative loss functions cannot deal with the ink bleeding properly, but the boundary-based loss functions erase ink bleeds to some extent in row 3.

Rows 4 and 5 show inconsistent hand pressure during printing, causing faded print. All loss functions demonstrate the difficulty in accurately determining stroke boundaries for faint print. The boundary-based loss functions tend to produce thicker strokes. The region-based loss functions produce many false positive dots. Because the Hausdorff distance calculates a distance for the extreme case, the boundary distance is effectively represented for convex shapes but not for complex shapes. Therefore, the result of $L_D^{10} \rightarrow L_H$ is very noisy within complex stroke arrangements.

Rows 6 and 7 show uneven ink coverage cases, including a streak and large holes. The region-based loss functions tend to generate holes. Row 8 contains the outer line of the typesetting board that needs to be removed. The boundary-based loss functions remove the line much better than the region-based loss functions. However, all the functions cannot completely remove the outer line since it is similar to a normal stroke. Row 9 demonstrates a case where GT was made incorrectly. The stroke in the enlarged part is misprinted. That is, the corresponding Korean character does not actually have the stroke. The worker labeling GT data did not completely remove the region generated by a pretrained segmentation model. All the models detect stroke-like large regions.

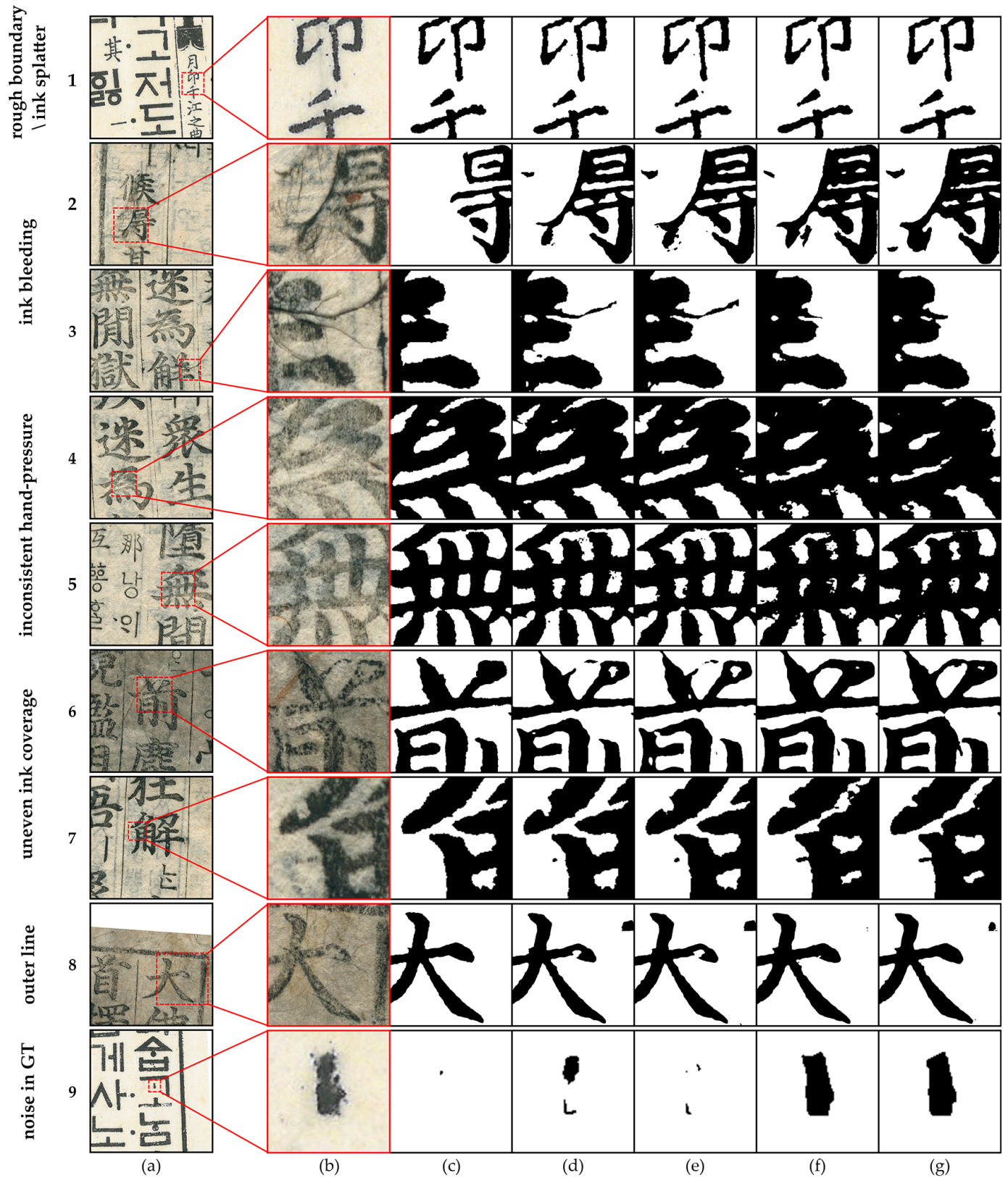


Figure 9. Example of character extraction with various loss functions. (a) Input image. (b) Part of the input image. (c) GT. (d) L_D . (e) L_{CE} . (f) $L_D^{10} \rightarrow L_H$. (g) $L_D^{10} \rightarrow L_{BGD}$.

Figure 10 demonstrates the segmentation results of various combinations of loss functions. As in Figure 9, Figures 9a–c show input images, enlarged problematic parts, and the corresponding GT labels, respectively. In general, combinations of loss functions become worse than using a single loss function in the objective performance (shown in Table 4), but visualization of the results indicates that the combined usages of loss functions improve segmentation quality and give more satisfactory results.

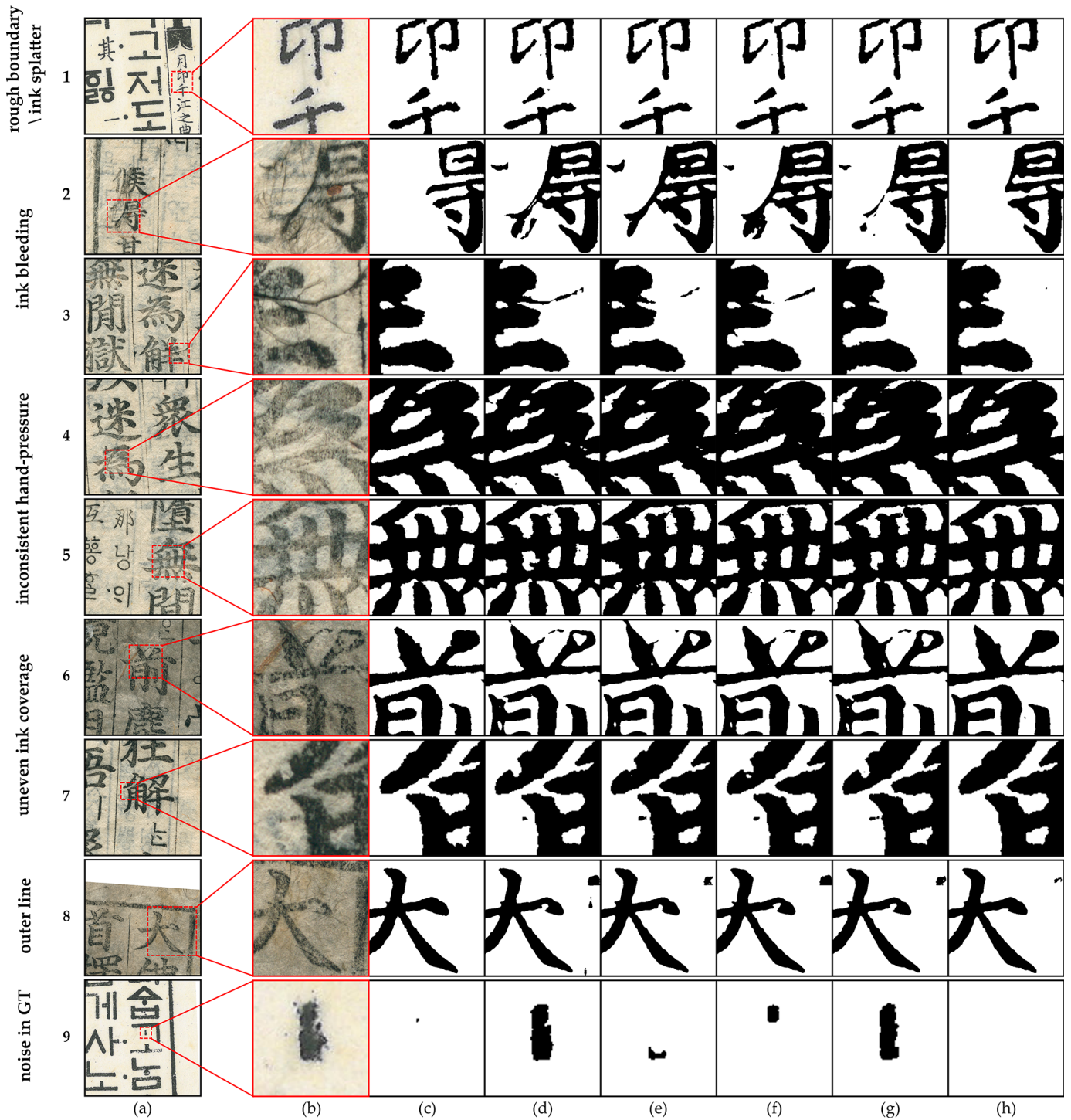


Figure 10. Example of character extraction with various loss functions. (a) Input image. (b) Part of the input image. (c) GT. (d) $L_D + L_{CE}$. (e) $L_D + L_H$. (f) $L_D + L_{GD}$. (g) $L_D + L_{BGC}$. (h) $L_D + L_{BGD}$.

In row 1, $L_D + L_{CE}$ still produces small dots, while the others remove them completely. It is worth noting that the proposed loss combination $L_D + L_{BGD}$ generates strokes with very smooth boundaries. Only the proposed one eliminates most of the ink bleeding in row 2. Unlike when using a single loss, the stroke thickness is maintained when using losses together in rows 4 and 5. The proposed loss combination produced significantly clean and smooth boundary strokes.

When the ink coverage was uneven, only the proposed loss combination filled the large holes within strokes and eliminated ink splatter noise in row 7. In contrast to Figure 9, where the two upper vertical strokes are connected through a stripe, the stripe was clearly removed due to the Dice loss.

In row 8, only the proposed one filled the large hole within the stroke and almost removed the outer line. The misprint in row 9 completely disappeared only when using the proposed loss combination. In summary, the proposed loss combination provides very satisfactory results by reducing noise, smoothing stroke outlines, and filling holes inside strokes.

In order to compare the generalization performance, we tested images from “Jikji”, the first book in Table 1, that were not used to build the dataset. “Jikji” is the oldest existing book printed with movable metal type in the world (A.D. 1377) and is currently held at the Bibliothèque Nationale de France. It predates Gutenberg’s Bible by 78 years and other books in Table 1 by over 70 years.

Figure 11 shows one page of “Jikji” and the segmentation results of the models trained with various combinational loss functions. Since there is no GT label for “Jikji”, we compared the results qualitatively. Due to the different compositions of the paper materials, the paper color of “Jikji” is much more red compared to the paper colors in the training dataset. In addition, there are many annotations—in particular, the brown strokes below some characters sometimes overlap with the lower strokes of the characters. Despite the different paper color, the models extracted most characters to some degree.

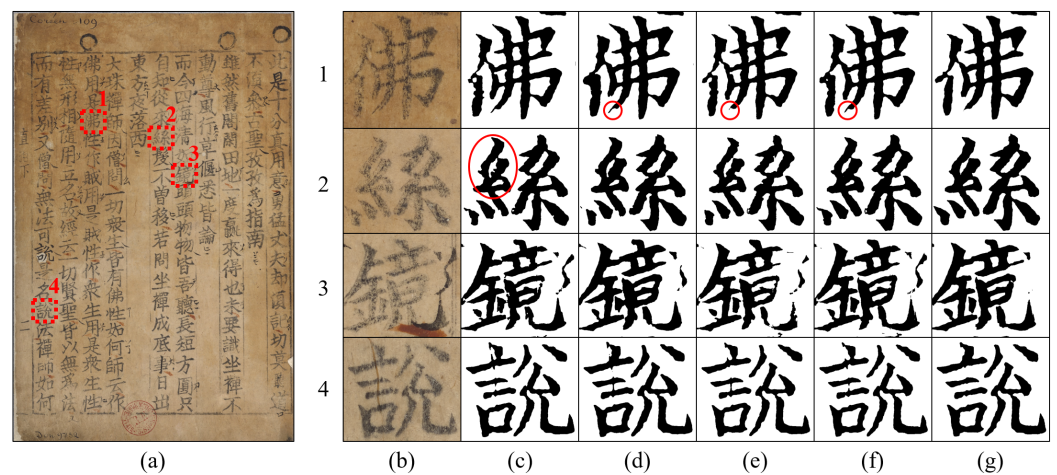


Figure 11. Generalization performance comparison of various loss functions. (a) One page of “Jikji”. Unlike the dataset for training the models, the paper in the scanned image looks more red. (b) Cropped character images of some printing problems, including (1) inconsistently printed, (2) double-printed character, (3) right black annotation and lower brown stroke, (4) long fiber-pulp. (c) $L_D + L_{CE}$. (d) $L_D + L_H$. (e) $L_D + L_{GD}$. (f) $L_D + L_{BGC}$. (g) $L_D + L_{BGD}$.

The image of character 1 was printed inconsistently and the left vertical strokes look faint. The results for $L_D + L_{CE}$, $L_D + L_{GD}$, and $L_D + L_{BGC}$ show holes within the left vertical stroke and rough stroke boundaries, while those for $L_D + L_H$ and $L_D + L_{BGD}$ depict smooth boundaries without holes. In addition, the end of the stroke marked with circles becomes too thin in the results for $L_D + L_{CE}$, $L_D + L_H$, and $L_D + L_{BGC}$.

Character 2 appears to be double printed. Similar to character 1, the results for $L_D + L_{CE}$, $L_D + L_{GD}$, and $L_D + L_{BGC}$ have holes within the stroke. The boundaries of the

two strokes within the red circle in (c) are too bumpy and the two strokes become wrongly connected in $L_D + L_{CE}$, $L_D + L_H$, and $L_D + L_{BGC}$.

The character in image 3 contains a black annotation in the right side and a brown stroke that were probably drawn later than the book's production. The proposed loss function removed the brown stroke successfully and the black annotation sufficiently, while others struggled to remove the black annotation. $L_D + L_{CE}$ and $L_D + L_{GD}$ produced rough and noisy strokes. In the character in image 4, a long fiber-pulp is noticeable. The proposed loss function perfectly eliminated it, while the fiber traces remained in all other results.

The segmentation models generalize well to the books from other eras with different characteristics; particularly, the model using the proposed loss achieves excellent segmentation results against various printing problems.

5. Conclusions

In this paper, we have proposed the boundary Gaussian distance loss function, a significant advancement for enhancing character segmentation in high-resolution scans of old Korean printed documents. Our loss function diverges from traditional metrics by incorporating Gaussian blur, which effectively smoothens character boundaries and reduces noise and outliers, markedly improving segmentation clarity and accuracy. Our experimental results demonstrate the superiority of our proposed loss function, especially in comparison with existing methods. The application of the boundary Gaussian distance loss yielded higher Dice scores for low-quality input images, signifying a notable improvement in character segmentation accuracy. The boundary Gaussian distance loss function is a valuable contribution to digital image processing, especially for historical document analysis.

By estimating and the typeface of each movable metal type with the proposed method, we can restore 3-D models of movable metal types. This will contribute to the study of the movable metal-type printing techniques of the time. For example, the amount of movable metal type produced at the time can be estimated by comparing the typeface of each metal type. Furthermore, its potential applicability to other languages and scripts opens new avenues for research in document digitization and preservation, making it a versatile tool for historical and linguistic studies.

Author Contributions: Conceptualization, K.-S.C.; methodology, K.-S.C.; software, K.-S.C. and W.-S.L.; validation, W.-S.L.; formal analysis, K.-S.C. and W.-S.L.; investigation, W.-S.L.; resources, W.-S.L.; data curation, W.-S.L.; writing—original draft preparation, W.-S.L.; writing—review and editing, K.-S.C.; visualization, W.-S.L.; supervision, K.-S.C.; project administration, K.-S.C.; funding acquisition, K.-S.C. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the Basic Science Research Program through the National Research Foundation of Korea (NRF), funded by the Ministry of Education (2022R1F1A1076305).

Data Availability Statement: The datasets presented in this article are not readily available because they are secure and private. Requests to access the datasets should be directed to book0496@koreatech.ac.kr.

Acknowledgments: We appreciate the support by Basic Science Research Program through the National Research Foundation of Korea (NRF) and the BK-21 FOUR program through NRF, under the Ministry of Education.

Conflicts of Interest: The authors declare no conflicts of interest. The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript; or in the decision to publish the results.

Abbreviations

The following abbreviations are used in this manuscript:

BGC	boundary-gradient consistency
BGD	boundary Gaussian distance
CE	cross-entropy
DSC	Dice score
GD	gradient difference
GT	ground truth

References

- Jeong, B.C.; Choi, K.S. 3-D Movable Type Reconstruction from Old Printed Documents using Deep Learning-based Character Extraction and Recognition. *J. Inst. Electron. Eng. Korea* **2022**, *59*, 74–83.
- Neudecker, C.; Baierer, K.; Federbusch, M.; Boenig, M.; Würzner, K.; Hartmann, V.; Herrmann, E. OCR-D: An End-to-End Open Source OCR Framework for Historical Printed Documents. In Proceedings of the International Conference on Digital Access to Textual Cultural Heritage, Brussels, Belgium, 8–10 May 2019; Association for Computing Machinery: New York, NY, USA, 2019; pp. 53–58. [\[CrossRef\]](#)
- Woo, J. A Study of Engraver’s Activity of Chosun Period Recorded in the Confucian Printing Woodblocks Kept in Advanced Center for Korean Studies. *J. Inst. Bibliogr.* **2019**, *79*, 89–110. [\[CrossRef\]](#)
- Lee, S. An Analysis of Movable Metal Types and Type-Setting in Jikji. *J. Inst. Bibliogr.* **2007**, *38*, 377–411. [\[CrossRef\]](#)
- Ok, Y. A Study on the Korean Metal Type Excavated from the Historic Site of Insa-dong. *J. Inst. Bibliogr.* **2023**, *93*, 31–50.
- Kim, D.K.; Ahmed, M.; Choi, K.S. Estimating the number of chases used for printing books with movable metal types. In *Eurographics Workshop on Graphics and Cultural Heritage*; The Eurographics Association: Eindhoven, The Netherlands, 2023. [\[CrossRef\]](#)
- Gedraite, E.; Hadad, M. Investigation on the effect of a Gaussian Blur in image filtering and segmentation. In Proceedings of the ELMAR-2011, Zadar, Croatia, 14–16 September 2011; pp. 393–396.
- Chamchong, R.; Fung, C. Character Segmentation from Ancient Palm Leaf Manuscripts in Thailand. In Proceedings of the Workshop on Historical Document Imaging and Processing, Beijing, China, 16–17 September 2011; Association for Computing Machinery: New York, NY, USA, 2011; pp. 140–145. [\[CrossRef\]](#)
- Kim, Y. Idu Script and its Chinese Version. *Soonchunhyang J. Humanit.* **2010**, 87–107.
- Kim, M.; Cho, K.; Kwag, H.; Kim, J. Segmentation of Handwritten Characters for Digitalizing Korean Historical Documents. In *International Workshop on Document Analysis Systems*; Springer: Berlin/Heidelberg, Germany, 2004; Volume 3163, pp. 114–124. [\[CrossRef\]](#)
- Shi, Y.; Peng, D.; Liao, W.; Lin, Z.; Chen, X.; Liu, C.; Zhang, Y.; Jin, L. Exploring ocr capabilities of gpt-4v (ision): A quantitative and in-depth evaluation. *arXiv* **2023**, arXiv:2310.16809.
- HoangVan, X.; TranQuang, P.; DinhBao, M.; VuHuu, T. Developing an OCR Model for Extracting Information from Invoices with Korean Language. In Proceedings of the International Conference on Advanced Technologies for Communications (ATC), Da Nang, Vietnam, 19–21 October 2023; pp. 84–89. [\[CrossRef\]](#)
- Liu, Y.; Li, Z.; Li, H.; Yu, W.; Huang, M.; Peng, D.; Liu, M.; Chen, M.; Li, C.; Jin, L.; et al. On the hidden mystery of OCR in large multimodal models. *arXiv* **2023**, arXiv:2305.07895.
- Rahman, A.; Ghosh, A.; Arora, C. UTRNet: High-Resolution Urdu Text Recognition in Printed Documents. In *International Conference on Document Analysis and Recognition*; Springer: Berlin/Heidelberg, Germany, 2023; pp. 305–324.
- Augustat, C.; Kapfhammer, W. Looking back ahead: A short history of collaborative work with indigenous source communities at the Weltmuseum Wien. *Bol. Mus. Paraen. Emilio Goeldi. Ciênc. Human.* **2017**, *12*, 749–764. [\[CrossRef\]](#)
- Droby, A.; Kurar Barakat, B.; Alaasam, R.; Madi, B.; Rabaev, I.; El-Sana, J. Text Line Extraction in Historical Documents Using Mask R-CNN. *Signals* **2022**, *3*, 535–549. [\[CrossRef\]](#)
- Mohammadian, M.; Maleki, N.; Olsson, T.; Ahlgren, F. Persis: A Persian Font Recognition Pipeline Using Convolutional Neural Networks. In Proceedings of the International Conference on Computer and Knowledge Engineering, Mashhad, Iran, 17–18 November 2022; pp. 196–204. [\[CrossRef\]](#)
- Yan, F.; Lan, X.; Zhang, H.; Li, L. Intelligent Evaluation of Chinese Hard-Pen Calligraphy Using a Siamese Transformer Network. *Appl. Sci.* **2024**, *14*, 2051. [\[CrossRef\]](#)
- Yan, F.; Zhang, H. SMFNet: One Shot Recognition of Chinese Character Font Based on Siamese Metric Model. *IEEE Access* **2024**, *12*, 38473–38489. [\[CrossRef\]](#)
- Tang, Y.; Wu, X. Scene text detection and segmentation based on cascaded convolution neural networks. *IEEE Trans. Image Process.* **2017**, *26*, 1509–1520. [\[CrossRef\]](#) [\[PubMed\]](#)
- Zhao, H.; Shi, J.; Qi, X.; Wang, X.; Jia, J. Pyramid scene parsing network. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 2881–2890.
- Bonechi, S.; Bianchini, M.; Scarselli, F.; Andreini, P. Weak supervision for generating pixel-level annotations in scene text segmentation. *Pattern Recognit. Lett.* **2020**, *138*, 1–7. [\[CrossRef\]](#)

23. Xu, X.; Zhang, Z.; Wang, Z.; Price, B.; Wang, Z.; Shi, H. Rethinking text segmentation: A novel dataset and a text-specific refinement approach. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Nashville, TN, USA, 20–25 June 2021; pp. 12045–12055.
24. Zhang, Z.; Sabuncu, M. Generalized Cross Entropy Loss for Training Deep Neural Networks with Noisy Labels. *arXiv* **2018**, arXiv:1805.07836.
25. Sudre, C.; Li, W.; Vercauteren, T.; Ourselin, S.; Cardoso, M. Generalised Dice overlap as a deep learning loss function for highly unbalanced segmentations. In Proceedings of the International Workshop on Deep Learning in Medical Image Analysis, Québec City, QC, Canada, 14 September 2017; pp. 240–248.
26. Mathieu, M.; Couprie, C.; LeCun, Y. Deep multi-scale video prediction beyond mean square error. *arXiv* **2015**, arXiv:1511.05440.
27. Mahendran, A.; Vedaldi, A. Understanding deep image representations by inverting them. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Columbus, OH, USA, 23–28 June 2014; pp. 5188–5196.
28. Lee, W.S.; Choi, K.S. Improvement of a Segmentation Network for Character Stroke Extraction from Metal Movable Type Printed Documents. *J. Inst. Electron. Eng. Korea* **2023**, *60*, 31–38.
29. Huttenlocher, D.; Klanderman, G.; Rucklidge, W. Comparing images using the Hausdorff distance. *IEEE Trans. Pattern Anal. Mach. Intell.* **1993**, *15*, 850–863. [[CrossRef](#)]
30. Wijethilake, N.; Kujawa, A.; Dorent, R.; Asad, M.H.; Oviedova, A.; Vercauteren, T.; Shapey, J. Boundary Distance Loss for Intra-/Extra-meatal Segmentation of Vestibular Schwannoma. In Proceedings of the International Workshop on Machine Learning in Clinical Neuroimaging, Singapore, 18 September 2022; pp. 73–82.

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.