Unveiling Text in Challenging Stone Inscriptions: A Character-Context-Aware Patching Strategy for Binarization[*]

Pratyush Jena, Amal Joseph, Arnav Sharma, Ravi Kiran Sarvadevabhatla
(pratyush.jena,amal.joseph,arnav.sharma)@research.iiit.ac.in,ravi.kiran@iiit.ac.in
Center for Visual Information Technology, International Institute of Information Technology, Hyderabad
Hyderabad, Telangana, India

Abstract

Binarization is a popular first step towards text extraction in historical artifacts. Stone inscription images pose severe challenges for binarization due to poor contrast between etched characters and the stone background, non-uniform surface degradation, distracting artifacts, and highly variable text density and layouts. These conditions frequently cause existing binarization techniques to fail and struggle to isolate coherent character regions. Many approaches sub-divide the image into patches to improve text fragment resolution and improve binarization performance. With this in mind, we present a robust and adaptive patching strategy to binarize challenging Indic inscriptions. The patches from our approach are used to train an Attention U-Net for binarization. The attention mechanism allows the model to focus on subtle structural cues, while our dynamic sampling and patch selection method ensures that the model learns to overcome surface noise and layout irregularities. We also introduce a carefully annotated, pixel-precise dataset of Indic stone inscriptions at the character-fragment level. We demonstrate that our novel patching mechanism significantly boosts binarization performance across classical and deep learning baselines. Despite training only on single script Indic dataset, our model exhibits strong zero-shot generalization to other Indic and non-indic scripts, highlighting its robustness and script-agnostic generalization capabilities. By producing clean, structured representations of inscription content, our method lays the foundation for downstream tasks such as script identification, OCR, and historical text analysis. Project page: https://ihdia.iiit.ac.in/shilalekhya-binarization/

CCS Concepts

• **Computing methodologies** → **Image segmentation**; **Computer vision problems**; Machine learning approaches; • **Applied computing** → *Digital libraries and archives.*

Keywords

stone inscriptions, binarization, document image analysis, deep learning, Indic scripts, epigraphy, historical documents

ACM Reference Format:
Pratyush Jena, Amal Joseph, Arnav Sharma, Ravi Kiran Sarvadevabhatla. 2025. Unveiling Text in Challenging Stone Inscriptions: A Character-Context-Aware Patching Strategy for Binarization. In *Indian Conference on Computer Vision, Graphics, and Image Processing (ICVGIP 2025), December 17–20, 2025, Mandi, India.* ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/3774521.3774539

1 Introduction

Stone inscriptions are rich sources of historical and linguistic knowledge, but their automated analysis remains challenging. Unlike scanned documents or manuscripts, they often exhibit severe degradation—shallow etching, erosion, surface noise, and uneven lighting. Text layout varies widely, and inscriptions frequently include decorative or non-textual elements, making standard image processing unreliable.

Binarization is a key step for text detection and document understanding. Traditional methods like Otsu [18] and Sauvola [21] often fail on degraded inscriptions with shallow or worn characters. Deep learning models such as U-Net [20] offer improved performance but rely on annotated data and fine-grained spatial understanding. Patching is commonly used to handle high-resolution images and varying text densities, yet often treated as a trivial detail. We argue that patch design is critical: poorly chosen patches can lack context or visual representativeness, degrading predictions. A principled patching strategy can significantly improve performance at minimal cost.

In this work, we focus on pixel-precise, character-level binarization for challenging stone inscriptions. We introduce a pipeline that uses an Attention U-Net trained with a patching strategy tailored to the inscription dimensions and typical character heights. This ensures each patch contains sufficient context for the model to distinguish foreground characters from the noisy stone background and non-textual carvings. To support training and evaluation, we construct a high-quality dataset of 203 annotated stone inscriptions with character fragments labeled at fine granularity. Although trained solely on one Indic script, our model generalizes well to other Indic and non-Indic scripts, demonstrating robustness in zero-shot scenarios. Our method outperforms both traditional and learning-based baselines, and incorporating our patching strategy also improves baseline performance.

Beyond solving a difficult binarization task, our approach lays a strong foundation for downstream epigraphic analysis, including script identification, OCR, and semantic region interpretation.

[*] Produces the permission block, and copyright information

ICVGIP 2025, December 17–20, 2025, Mandi, India
© 2025 Copyright held by the owner/author(s).
ACM ISBN 979-8-4007-1930-1/25/12
https://doi.org/10.1145/3774521.3774539

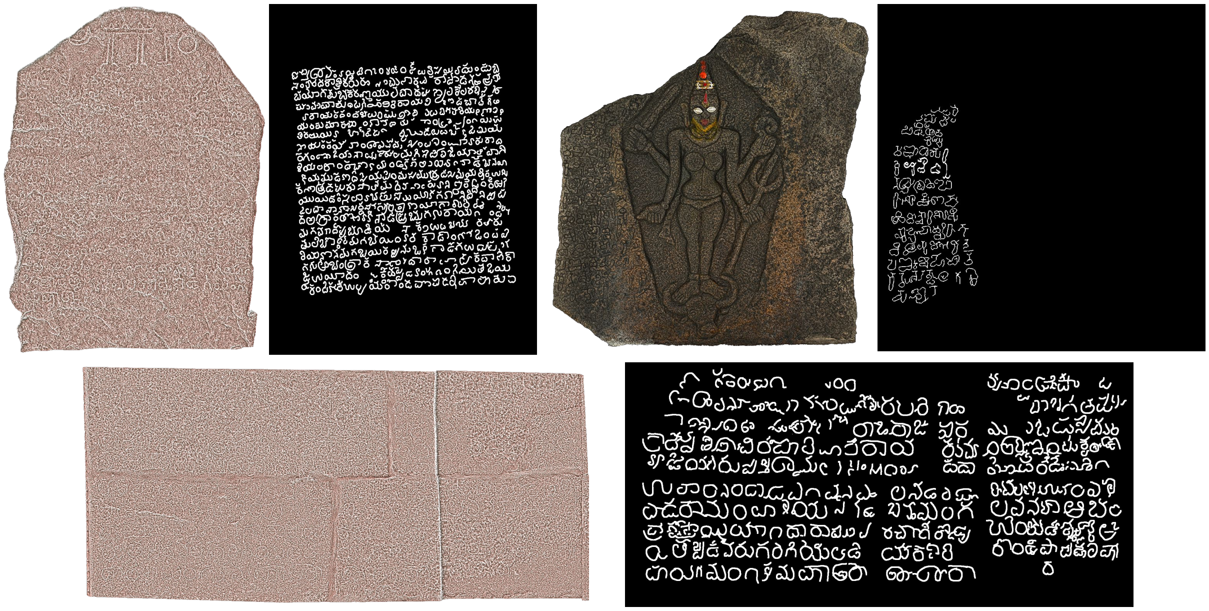

Figure 1: Sample images and their corresponding binarization ground truth from the annotated stone inscription dataset. Notice the difficulty distinguishing the shallow handwritten text etching from the background stone texture with naked eye.

2 Related Work

Document Image Binarization: Classical methods like Otsu [18], Niblack, and Sauvola [15, 21] rely on global or local intensity thresholds. Though efficient, they struggle with degraded or textured backgrounds typical of stone inscriptions. Contrast-enhancement-based methods [24], and MRF-based formulations like Howe's [8] offer more robustness but still under perform on complex cases.

Deep learning methods have led to significant improvements in binarization. U-Net [20] and its variants (including Attention U-Net [17]), as well as Fully Convolutional Networks optimized for document binarization [25], have shown strong performance in pixel-wise segmentation. The DIBCO 2019 [19] leader board represents the current state of the art. NAF-DPM [6] utilizes a diffusion probabilistic model while [31] follows a gated convolutional architecture. [5, 23] follow the Transformer [28] architecture.

Adaptive Patching Mechanisms: Patching is widely used in document analysis to handle high-resolution images and augment training data. Most approaches adopt a fixed patch size with overlap [6, 23, 27], which works well for uniform layouts. However, in stone inscriptions where character size and spacing vary significantly, fixed-size patching often fails—either missing entire characters or including excessive background, weakening the model's focus.

Adaptive strategies have been explored, but are not well-suited for epigraphy. LineTR [2] adapts patch size based on interline gaps, which are inconsistent or absent in inscriptions. Quad-tree decomposition [10] leads to overly small patches with poor context, and spatially-adapted sliding windows [7, 11] struggle to maintain coherence across scales. These limitations motivate our custom patching strategy tailored to the structure and variability of stone inscriptions.

Epigraphy and Stone Inscription Analysis: Computational work on stone inscriptions across different scripts remains limited. A recent overview of computational epigraphy surveys modern approaches including template matching and edge-based filtering [9]. For Indian scripts, HOG+SVM has been used to recognize ancient Tamil inscription characters [4], while CNN-based OCR for Ashokan Brahmi demonstrates strong performance using transfer learning [3]. Adhikari and Palaniappan [1] proposed a deep-learning pipeline for segmenting and classifying symbols in Indus script seal impressions. In Greek and Roman studies, image enhancement and template matching have been applied to recover faint carvings from weathered surfaces [32]. Munivel et al. [13] propose a multi-level binarization technique for Tamil inscriptions, but report limited robustness under severe degradation.

End-to-end recognition systems often assume clean segmentation and high contrast, which is rarely the case in inscriptions. Erosion, poor lighting, and fragmented layouts degrade recognition performance. Our method instead focuses on robust binarization, forming a reliable, modular foundation for OCR and script analysis across scripts and degradation levels.

3 Dataset

We introduce a new dataset of 203 high-resolution images of stone inscriptions carved in the script. The inscriptions span diverse historical periods, styles of etching, and physical conditions, including erosion, moss growth, cracks, and inconsistent lighting. These stone inscriptions are extremely challenging as the etching strokes and the background surface are visually indistinguishable from noise. Along with text, there are visual elements like reliefs and iconography which further increase the complexity. Each image has been carefully annotated at the character-fragment level, resulting in fine-grained binary masks for every distinguishable fragment (See Fig. 1).

The inscription images used in this work were obtained from the Akshara Bhandara digital archive hosted by the Mythic Society [14]. All images are part of the Wikimedia Commons [29] public domain collection under appropriate licensing. We curated a subset of these images for annotation, prioritizing diversity in layout, surface quality, and script styles.

Annotations were created using *GIMP* [26] in overlay mode with a layered brush-based approach. Annotators used freehand brush tools, aided by XP-Pen drawing tablets [30] for precise stroke control. Annotators were instructed to trace the centerline of strokes and fill the character body. This setup allowed accurate delineation of even shallow or partial etchings under low contrast. Depending on complexity, annotation time ranged from 15 to 20 minutes for clean, low-density images to over 2 hours for eroded, dense inscriptions. The dataset was annotated over a 1 month period by a team of 4 annotators.

The dataset is divided into training (85%), and test (15%) splits, stratified to reflect diversity in etching quality, surface texture, and text layout. Some salient statistics of our dataset can be seen in Table 1.

Table 1: Dataset Statistics (203 stone inscription images)

Statistic	Min	Max
Character fragments per image	1	708
Image width (pixels)	351	3840
Image height (pixels)	148	2784
Aspect Ratio	0.34	13.3

4 Proposed Approach

We propose a novel, spatially adaptive, Character-Context-Aware patching mechanism (Sec. 4.1). The resulting patches are used to train a binarization network (Sec. 4.2). At test time, a self-refining inference pipeline (Sec. 4.3) is used to intelligently mimic the training-time strategy, thereby enabling robust binarization.

4.1 Character-Context-Aware Patching

The foundation of our patching approach is a novel three-step strategy for generating training patches. The central idea is to use the character component itself as the fundamental unit of measurement, thereby creating a system that is inherently adaptive to the specific content of each image. Refer to Fig. 2 for blue circled items below.

4.1.1 *Step 1: Optimal Character Height Calculation* ❶

To make our pipeline adaptive to image content, we first calculate $\overline{h}_{cc}$ - a single, robust value representing the average character height. This step is critical since it allows all subsequent operations to be scale-invariant. We begin by identifying all connected components in the binary ground-truth mask and compute their heights $h_{\text{cc},i}$ where i indexes the components. To ensure that $\overline{h}_{cc}$ is not skewed by tiny components like diacritics or large non-textual elements (e.g. decorative carvings), we retain only those components whose heights fall within the inter-quartile range (IQR), i.e. $\mathcal{H}_{\text{IQR}} = \left\{ h^i_{cc} \,\middle|\, Q^h_1 \leq h^i_{cc} \leq Q^h_3 \right\}$ where Q^h_1 is the 25th height percentile and Q^h_3 is the 75th percentile. This statistical trimming isolates the main body of characters. $\overline{h}_{cc}$ is then computed as the mean height of this robust, filtered set, i.e. $\overline{h}_{cc} = \frac{1}{|\mathcal{H}_{\text{IQR}}|} \sum_{h^i_{cc} \in \mathcal{H}_{\text{IQR}}} h^i_{cc}$.

4.1.2 *Step 2: Identify Foreground/Background Region* ❷

With $\overline{h}_{cc}$ established, we next partition the image into a foreground region (containing text) and a background region (containing only stone texture, noise, and other non-textual elements). This explicit separation allows for targeted sampling, ensuring the model is exposed to a rich and balanced set of both positive (text) and hard-negative (visually similar noise) examples. The foreground region is identified via a two-stage morphological dilation of the character components bounding boxes, where the kernel dimensions are adaptive, scaling proportionally with $\overline{h}_{cc}$. The morphological operation is done using kernel size of height of $s_1 \times \overline{h}_{cc}$ and width of $s_2 \times \overline{h}_{cc}$ for stage 1. The kernel dimensions are interchanged for stage 2. This makes the process robust to variations in character spacing and scale. It also ensures spaces between characters cannot be taken as background. The background region is the complement of this final dilated foreground area.

4.1.3 *Step 3: Multi-Scale Patch Sampling* ❸

Finally, we extract training patches from the identified regions. To remove outlier components using a height-based criteria, we define $C_{\text{valid}} = \left\{ c \in C \;\middle|\; Q^h_1 - q_1 \,\text{IQR}_h \leq h^i_{cc} \leq Q^h_3 + q_2 \,\text{IQR}_h \right\}$. We then compute a preliminary foreground count $N'_{\text{fg}} = |C_{\text{valid}}| \, R_{\text{base}}$ where R_{base} is the base sampling rate for the expected number of patches per valid character. We clamp the count to obtain the final foreground count $N_{\text{fg}} = \max\big(N_{\text{min}}, \; \min(N'_{\text{fg}}, \, N_{\text{max}})\big)$ where N_{min} is minimum number of patches and N_{max} maximum number of foreground patches. The clamping is crucial as it prevents images with extremely high text density to dominate the training set

We set a background patch count proportional to the available background area: $N_{\text{bg}} = \frac{A_{\text{bg}}}{A_{\text{total}}} N_{\text{bg}^{\text{max}}}$. The $N_{\text{bg}^{\text{max}}}$ enforces a max limit which prevents oversampling from a text sparse images. As the background regions are generally homogeneous and less information dense than text, even sparser sampling captures enough variability. This allows us to focus the computational and learning capacity for the textual region.

With the patch counts determined, the side length of each extracted patch, L_{patch}, is adaptively sized based on the mean character height $\bar{h}_{\text{cc}}$: $L_{\text{patch}} = k \cdot \bar{h}_{\text{cc}}$ where k is sampled from a uniform distribution. This multi-scale approach serves as a powerful form of data augmentation, making the model inherently robust to variations in character scale. Refer Fig. *3a* to view sample patches created by our patching method. Notice that the character scales relative to patch image length are consistent thanks to our patching method. Overall, the patches generated by our approach ensure the binarization model (next section) can be trained with consistent and representative views of both text and non-text regions.

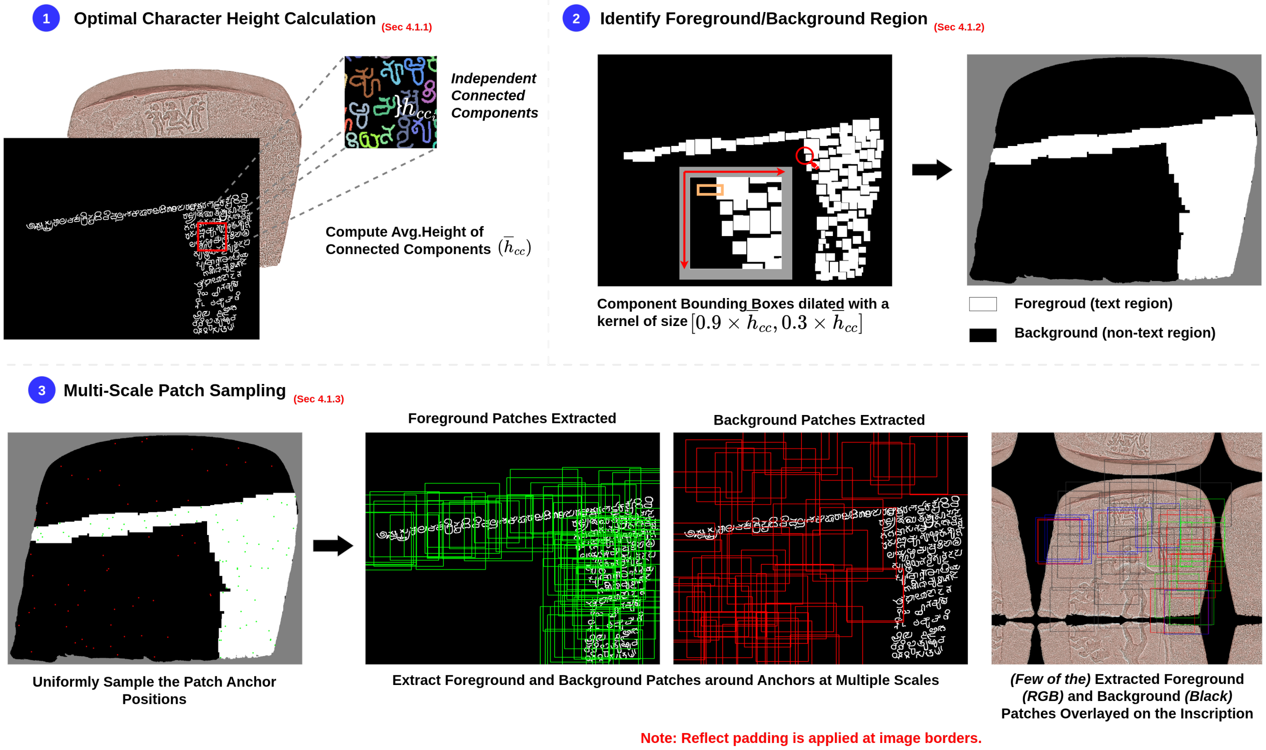
1 Optimal Character Height Calculation (Sec 4.1.1)
Independent Connected Components
Compute Avg.Height of Connected Components $(\bar{h}_{cc})$
2 Identify Foreground/Background Region (Sec 4.1.2)
Component Bounding Boxes dilated with a kernel of size $[0.9 \times \bar{h}_{cc}, 0.3 \times \bar{h}_{cc}]$
Foregroud (text region)
Background (non-text region)
3 Multi-Scale Patch Sampling (Sec 4.1.3)
Foreground Patches Extracted
Background Patches Extracted
Uniformly Sample the Patch Anchor Positions
Extract Foreground and Background Patches around Anchors at Multiple Scales
(Few of the) Extracted Foreground (RGB) and Background (Black) Patches Overlayed on the Inscription
Note: Reflect padding is applied at image borders.

Figure 2: Overview of our Character-Context-Aware Patch Selection Strategy.
① **First, we compute the mean character height $(\bar{h}_{cc})$ using connected components.** ***(Sec. 4.1.1)***
② **Next, dilate (with a kernel adaptive to $(\bar{h}_{cc})$) to identify *foreground (text)* and *background (non-text)* regions.** ***(Sec. 4.1.2)***
③ **Finally, uniformly sample anchor points from these regions to extract multi-scale patches.** ***(Sec. 4.1.3)***
This strategy ensures that each patch contains consistently-scaled context, enabling the model to effectively learn the distinction between character strokes and background noise.

4.2 Attention U-Net Binarizer

The patches obtained using our patching strategy from previous section are used to train an Attention UNet [17] model for the binarization task. Attention U-Net extends the U-Net [20] architecture by introducing attention gates at skip connections, allowing the model to focus on relevant spatial regions while suppressing irrelevant background noise. This mechanism proves especially useful in the context of inscriptions, where faint strokes and background texture are difficult to distinguish. Refer Fig. 4, 6.

4.3 Self-Refining Inference Pipeline

During inference, our goal is to apply the learned binarization model to new, unseen stone inscription images. To ensure the patches are scaled appropriately to the character heights and to replicate the patching strategy used during training, we introduce a two-stage, self-refining inference pipeline. This pipeline first generates a coarse prediction of the text regions and then uses this information to guide a more precise, context-aware binarization. Both the stages use the same trained Attention U-Net model (Sec. 4.2). Refer to Fig. 5 for circled numbers below.

4.3.1 ***Stage 1: Initial Prediction:*** ① We first perform a multi-scale prediction, where the image is processed using sliding windows of four different scales (*256*, *384*, *512* and *768* pixels). This ensures that at least one of the scale would be optimal for the characters present in the image. For each pixel, across all scales, the value (predicted from the model) with maximum probability is taken to create a preliminary binary map. This initial map, though potentially coarse, serves as a "pseudo-ground truth" mask, providing a strong prior for identifying foreground regions in *Stage 2*.

4.3.2 ***Stage 2: Context-Aware Refinement:*** ② By treating the output from *Stage 1* as a binary map ground truth, we apply our proposed *Character-Context-Aware Patching strategy* (Sec. 4.1). This allows for a more targeted and dense sampling of patches from the identified foreground (text) and background regions at their optimal scales. These newly sampled patches are then passed through the trained model a second time to yield the final, refined predictions. This refinement step is crucial as it leverages patches that are optimized for character context, leading to a significant reduction in false positives and an improvement in the coherence of the binarized text boundaries (Fig. 5). We employ a patch merging strategy

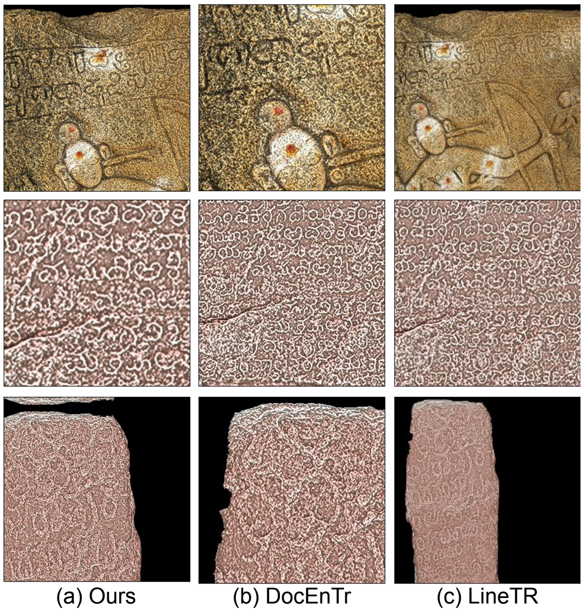

Figure 3: Our Character-Context-Aware Patching produce patches of good context, where the amount of textual information is consistent across the patches. Notice the character height is similar across the patches relative to the patch dimensions with our patching method.

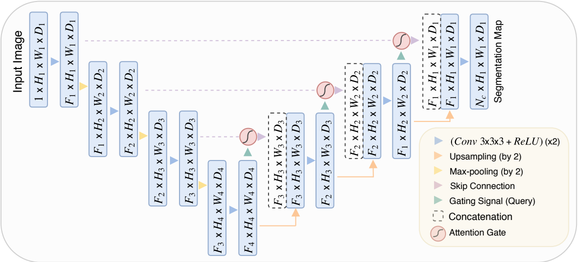
Input Image
$1 \times H_1 \times W_1 \times D_1$
$F_1 \times H_1 \times W_1 \times D_1$
$F_1 \times H_2 \times W_2 \times D_2$
$F_2 \times H_2 \times W_2 \times D_2$
$F_2 \times H_3 \times W_3 \times D_3$
$F_3 \times H_3 \times W_3 \times D_3$
$F_3 \times H_4 \times W_4 \times D_4$
$F_4 \times H_4 \times W_4 \times D_4$
$F_3 \times H_3 \times W_3 \times D_3$
$F_3 \times H_3 \times W_3 \times D_3$
$F_2 \times H_3 \times W_3 \times D_3$
$F_2 \times H_2 \times W_2 \times D_2$
$F_2 \times H_2 \times W_2 \times D_2$
$F_1 \times H_2 \times W_2 \times D_2$
$F_1 \times H_1 \times W_1 \times D_1$
$F_1 \times H_1 \times W_1 \times D_1$
$N_c \times H_1 \times W_1 \times D_1$
Segmentation Map
(*Conv* 3x3x3 + *ReLU*) (x2)
Upsampling (by 2)
Max-pooling (by 2)
Skip Connection
Gating Signal (Query)
Concatenation
Attention Gate

Figure 4: Architecture of the Attention U-Net used for patch-wise binarization. Attention gates modulate encoder features before concatenation at each decoder stage. Model architecture reproduced from [17] under CC BY 4.0 license.

to aggregate dense patch-level predictions into a complete binary map. Refer Algorithm 1 for more information.

This two-stage, self-refining process significantly enhances the final binarization quality. While the effectiveness of the second stage depends on the initial identification of text regions in the first stage, we found our multi-scale sliding window approach to be robust in practice, identifying the vast majority of foreground regions and enabling high-quality, refined predictions even in cases of hard zero-shot results, even in cases of different scripts (see Fig. 7).

5 Implementation Details

Data Preparation and Patching Hyperparameters: To implement our Character-Context-Aware Patching strategy (Sec.4.1), we calibrated a set of hyperparameters that balance text-region coverage, background sampling, and multi-scale augmentation. For the kernel in Step 1 of patching, we use $s_1 = 0.3, s_2 = 0.9$. For foreground sampling, we use a base rate $R_{\text{base}} = 0.5$, extracting 5 patches per 10 valid character component. In Step 2, we set IQR scaling factors $q_1 = q_2 = 1.5$. The total number of foreground patches is clamped between 10 ($N_{\min}$) and 250 ($N_{\max}$). For background (negative) sampling, the limit is 75 patches per image ($N_{\text{bg}}^{\max}$). To introduce scale variation, each patch's side length is set to $L_{\text{patch}} = k \cdot \bar{h}_{cc}$, with $k \sim \mathcal{U}(4, 12)$. Each patch is resized to 512×512 pixels before passing it to the network.

Loss Function for Attention U-Net: We selected a hybrid Dice-BCELoss function. This choice is motivated by its suitability for highly imbalanced segmentation tasks. The Binary Cross Entropy component provides stable pixel-level gradients while the Dice term directly optimizes the F1-score (segmentation overlap), encouraging the model to produce spatially coherent and complete character shapes. The Dice loss to BCE loss are equally weighted. The model is trained with Adam optimizer, with learning rate of 1×10^{-4}. We trained our model on a single Nvidia A6000 GPU with a batch size of 16 for 50 epochs while storing the checkpoints with the best Dice score.

Algorithm 1 Self-Refining Inference Pipeline

```
Require: Inscription image $I$, trained model $M$
Ensure: Final binarized map $B_{\text{final}}$

Stage ① : Initial Prediction (Coarse Map Generation)
$(H, W) \leftarrow \text{size}(I)$
$S \leftarrow \{256, 384, 512, 768\}$        ▷ Sliding window scales
$P_{\text{pyr}} \leftarrow \mathbf{0}^{|S| \times H \times W}$  ▷ Stores prediction maps for each scale in $S$
for all $s \in S$ do
    $(\text{Patches, Locs}) \leftarrow \text{SlidingWindow}(I, s)$
    $Y \leftarrow M(\text{Patches})$
    $P_{\text{pyr}}[s] \leftarrow \text{MergePatchPred}(Y,\ \text{Locs},\ H, W)$
end for
$P_{\text{coarse}} \leftarrow \max_s(P_{\text{pyr}}[s])$        ▷ Max-fusion over scales
$B_{\text{pseudo}} \leftarrow (P_{\text{coarse}} > 0.5)$   ▷ Generate pseudo-ground truth

Stage ② : Context-Aware Refinement
$\bar{h}_{\text{cc}} \leftarrow \text{CalcAvgIQRHeight}(B_{\text{pseudo}})$
$(\text{Patches}', \text{Locs}') \leftarrow \text{ContextAwarePatch}(I,\ B_{\text{pseudo}},\ \bar{h}_{\text{cc}})$
$A \leftarrow \mathbf{0}^{H \times W}$          ▷ Accumulator for binary logits
$C \leftarrow \mathbf{0}^{H \times W}$          ▷ Accumulator for counts
$Y' \leftarrow M(\text{Patches}')$
for all each $(y,\ \ell)$ in zip$(Y',\ \text{Locs}')$ do
    $A[\ell]$ += $y$
    $C[\ell]$ += 1
end for
$P_{\text{final}} \leftarrow A\ \oslash\ (C + \varepsilon)$   ▷ Average overlapping predictions
$B_{\text{final}} \leftarrow (P_{\text{final}} > 0.5)$
return $B_{\text{final}}$
```

6 Experiments

We evaluate our method on the test split of our inscription dataset using the standard document binarization metrics: Peak signal-to-noise ratio (*PSNR*), F-measure (*FM*), pseudo-F-measure (F_{ps}) [16]

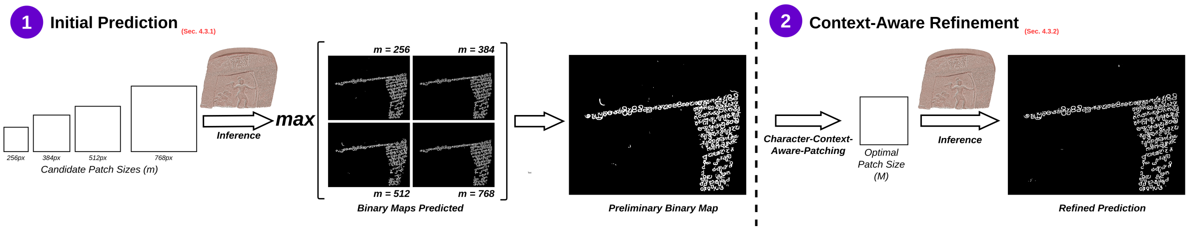

Figure 5: Overview of our Self-Refining Inference Pipeline.
1 First, we perform inference with patch sizes *256, 384, 512, 768* and fuse their predictions to get a preliminary binary map. *(Sec. 4.3.1)*
2 Using this map as a pseudo-ground truth, we then determine the optimal patch size and perform a final inference pass to get the refined prediction. *(Sec. 4.3.2*)

and Distance Reciprocal Distortion (*DRD*) [12]. We compare our binarization pipeline against both zero-shot and supervised baselines, and conduct ablation studies to assess the contribution of each component in our pipeline.

Binarization Baselines:
We consider the baselines outlined below.

- Otsu [18]: Global thresholding method that selects a threshold minimizing intra-class variance.
- Sauvola [21]: A local adaptive thresholding algorithm.
- Standard U-Net [20]: Encoder-decoder architecture with skip connections.
- FCN [25]: Fully convolutional network that replaces dense layers with up-sampling for semantic segmentation.
- NAF-DPM [6]: Based on Diffusion Probabilistic Model, the current SOTA on DIBCO 2019.

Patching Strategies
These baselines are evaluated on the following patching strategies

(a) Character-Context-Aware Patching ***(ours)***: Patch sizes are scaled to the character component height.
(b) Context-Adapted Patching *(LineTR [2])*: Patch sizes are scaled to the average interline gap between the text lines.
(c) Fixed-patching with 50% overlap *(DocEnTr [23])*

For Otsu [18] and Sauvola [21], patching is not applied as they operate on global or local thresholding principles that do not require division of the image into smaller regions.

7 Results

Table 2 presents a comparative analysis of all methods. Our method outperforms all baselines by a significant margin, both in the test set and in zero-shot setting. Our patching strategy performs the best when coupled with Attention UNet [20]. We can also observe a performance improvement when our patching strategy is coupled with other models as well. See Fig. 8 for qualitative comparison of binary map predictions.

7.1 Ablation Studies

We conduct ablation studies to isolate the contribution of each major component in our pipeline (see Table 3).
Attention gates in U-Net Architecture: Removing the attention gates leads to a noticeable drop in performance. Without attention, the model struggles to focus on relevant spatial regions. It becomes more easily confused by the complex textures of the stone, leading to incomplete character strokes and a higher rate of false positives in the output.

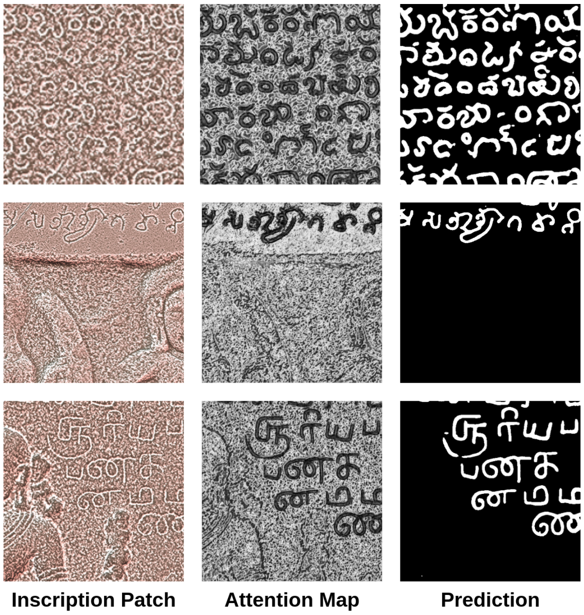

Figure 6: Attention maps extracted from the decoder layer of Attention U-Net reveal how the model learns to selectively target foreground text. Darkened regions correspond to areas the model focuses on during binarization. The model selectively enhances low-contrast strokes and suppresses background patterns.

Character-Context-Aware Patching Strategy: Eliminating our *Character-Context-Aware patching strategy* significantly affects the model's ability to distinguish characters from background artifacts. The F-Measure plummets from 72.20 to 41.94, and the DRD score worsens from 12.14 to 28.43 (see Table 3). This is the most significant performance drop in our ablations, validating the core premise of our paper.
Patch Size Multiplier: As shown in Table 4, smaller size multipliers resulted in patches that are too tightly cropped, depriving the model of the surrounding stone texture needed to distinguish faint strokes from noise and leading to fragmented predictions. Conversely, larger multipliers causes the character to become too small relative to the patch, diluting the learning signal and offering no additional performance benefits.

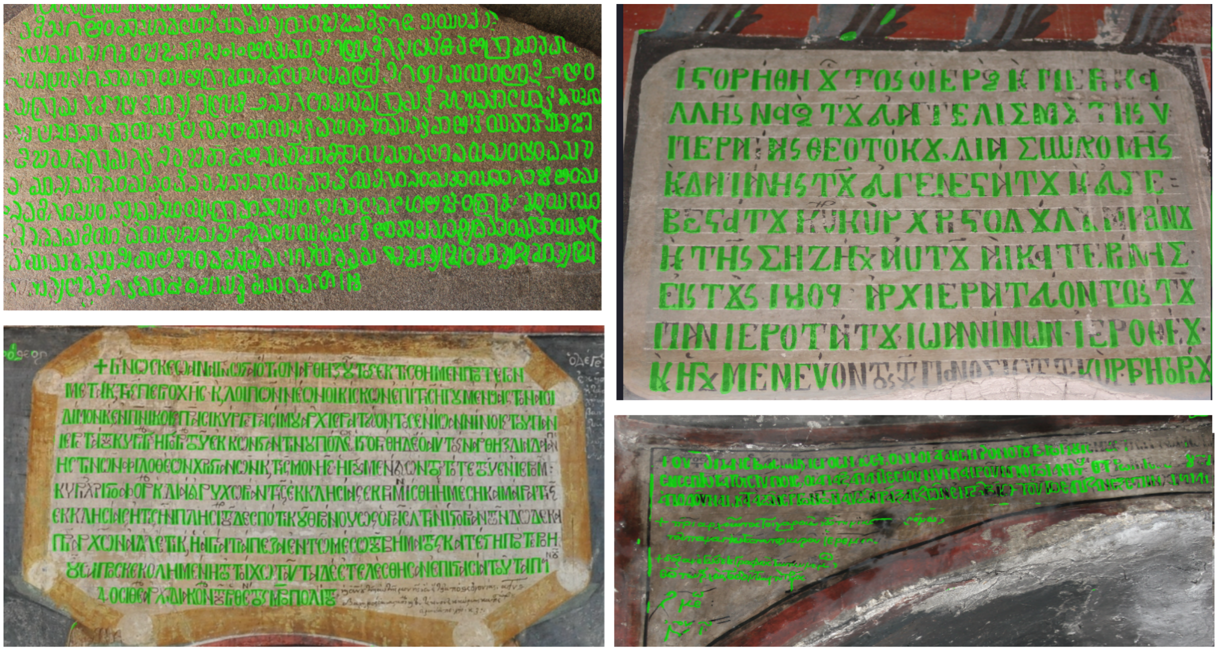

Figure 7: Demonstration of our model's robust zero-shot generalization. Examples are from challenging, in-the-wild Indic and Byzantine-era Medieval Greek [22] inscriptions. Despite significant variations in script, lighting, and surface degradation, our method consistently produces clean, legible binarizations. Note: The predicted binary maps are overlaid on the inscriptions.

Table 2: Comparison of models and patching strategies on Test Set and Zero-shot dataset.

Model	Patching	Test Set				Zero Shot			
		PSNR ↑	FM ↑	F_{ps} ↑	DRD ↓	PSNR ↑	FM ↑	F_{ps} ↑	DRD ↓
Otsu [18]	No Patching	3.12	7.48	6.78	354.15	4.77	33.38	34.54	102.59
Savoula [21]	No Patching	4.47	12.41	11.16	198.25	6.98	28.11	30.67	51.65
Standard U-Net [20]	LineTR [2]	13.45	48.27	53.34	19.20	8.63	9.52	10.76	27.30
	DocEnTr [23]	14.55	59.68	65.85	13.48	8.77	19.78	22.06	25.91
	ours	**14.71**	64.15	70.53	12.37	8.68	29.67	31.50	25.71
FCN [25]	LineTR [2]	12.64	38.41	41.47	24.62	8.41	11.24	11.79	28.63
	DocEnTr [23]	13.20	52.41	56.43	20.43	8.44	13.25	14.30	28.20
	ours	14.01	59.86	64.50	15.67	8.57	38.22	40.43	26.11
NAF-DPM [6]	LineTR [2]	13.77	39.42	45.85	17.96	8.54	1.87	2.09	38.56
	DocEnTr [23]	14.28	51.48	60.26	15.75	8.74	16.40	18.93	37.31
	ours	14.10	51.98	61.07	16.31	**9.08**	27.66	30.98	34.30
Attention U-Net [17]	LineTR [2]	13.50	48.56	53.30	18.79	8.62	9.62	10.82	27.40
	DocEnTr [23]	14.41	59.68	66.68	13.84	8.72	15.84	17.83	26.41
	ours	14.61	**66.03**	**72.20**	**12.14**	8.92	**39.68**	**42.30**	**23.59**

Multi-scale Patch Sampling: We evaluated whether using a range of patch scales leads to more robust models than training with a single fixed scale. First, we trained models with fixed patch size multipliers k to find the best-performing value (Table 4). Performance dropped for small ($k < 4$) and large ($k > 12$) scales. Next, we compared this fixed-scale model to our multi-scale strategy, where k is sampled uniformly from $[4, 12]$. As shown in Table 5, multi-scale approach outperforms the fixed baseline across all metrics, confirming that scale variation acts as strong data augmentation. By training across zoom levels and contexts, the model becomes more robust to the wide variability in real-world inscriptions.

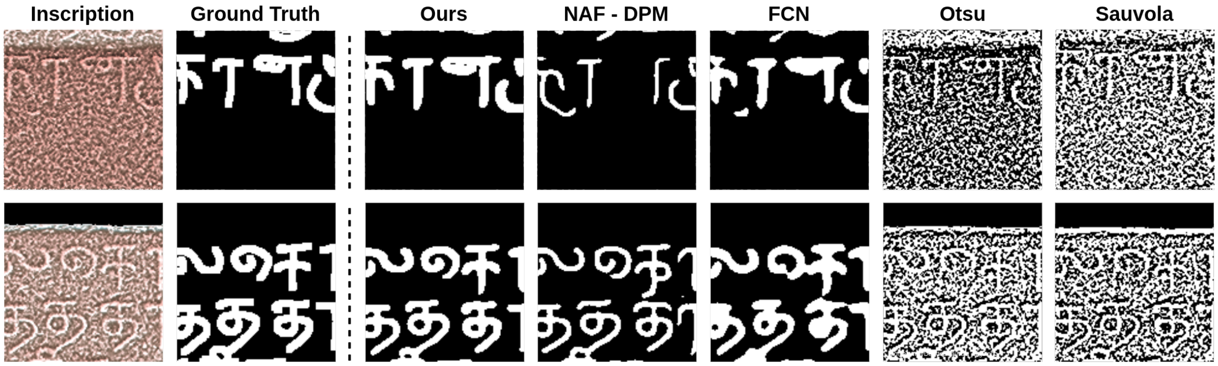

Figure 8: Qualitative Comparison between our method and other approaches. From left to right, the input inscription image and ground truth mask, and the predictions by Otsu [18], Savoula [21], FCN [25], NAF-DPM [6] and our model. The characters restored by our network are clearly more readable and accurate.

Table 3: Ablation study of model variants

Variant	PSNR ↑	FM ↑	F_{ps} ↑	DRD ↓
No Attention Gates	14.71	64.15	70.53	12.37
Without Patching	12.03	39.69	41.94	28.43
Ours (full pipeline)	14.61	66.03	72.20	12.14

Table 4: Ablation study on the fixed patch size multiplier (k). Each model is trained with a single, fixed patch size of $k \times \overline{h}_{cc}$. This study validates our choice of the optimal multiplier range.

Patch Size Multiplier (k)	PSNR ↑	FM ↑	F_{ps} ↑	DRD ↓
1.5	12.43	18.46	21.22	26.79
3.0	13.33	53.08	58.12	19.26
4.0	13.52	53.48	58.49	18.64
5.0	12.91	52.02	55.74	22.6
7.0	13.98	57.94	62.65	16.16
9.0	**14.2**	60.17	65.79	14.89
12.0	14.17	60.96	**66**	**14.69**
15.0	13.98	**61.32**	65.93	15.5

Table 5: Ablation study comparing our multi-scale patch sampling against the best-performing fixed-scale strategy. This demonstrates the benefit of training on patches of varying sizes.

Patching Strategy	PSNR ↑	FM ↑	F_{ps} ↑	DRD ↓
Fixed-Scale (best, $k = 12$)	14.17	60.96	66	14.69
Multi-Scale ($4 \leq k \leq 12$)	**14.61**	**66.03**	**72.20**	**12.14**

7.2 Zero-Shot Generalization to Other Indic Scripts

To assess the generalizability of our approach, we evaluate our model on Indic and non-Indic stone inscriptions, which were not seen during training. These images were captured in the wild using consumer-grade cameras and manually annotated by us. Despite notable visual and structural differences from the training data, our binarization model successfully extracts foreground strokes (See Fig . 7). These results suggest that our binarization model focuses on generic edge and shape cues, rather than script-specific features, demonstrating its potential for broader application in epigraphic analysis across diverse Indic scripts.

8 Conclusion

In this paper, we proposed a novel patching strategy and an Attention U-Net model tailored for pixel-precise binarization of challenging stone inscriptions. Our patching method generates *Character-Context-Aware* patches, that capture optimal amount of textual information, enabling the model to better distinguish character regions from background artifacts. Extensive experiments demonstrate that our method significantly outperforms both traditional and modern baselines, including the current state-of-the-art on the DIBCO 2019 benchmark. We also show that by adapting our patching mechanism, the performance of existing methods can be further improved.

Moreover, our model exhibits strong zero-shot generalization to unseen Indic and Western stone inscriptions, indicating that it learns script-agnostic structural patterns rather than language-specific features. These results highlight the potential of our approach as a robust preprocessing step for downstream tasks such as script identification, OCR, transliteration, and linguistic analysis across diverse ancient scripts.

More broadly, our work contributes to the development of robust tools for digital epigraphy and supports the large-scale computational study of South Asian textual heritage.

9 Acknowledgments

This work is supported by Ministry of Electronics and Information Technology (MeiTY), Government of India. We also acknowledge Akshara Bhandara for providing the dataset images used in this research.

References

[1] Ronojoy Adhikari and Satish Palaniappan. 2017. Deep Learning the Indus Script. *arXiv preprint arXiv:1702.00523* (2017). https://arxiv.org/abs/1702.00523

[2] Vaibhav Agrawal, Niharika Vadlamudi, Muhammad Waseem, Amal Joseph, Sreenya Chitluri, and Ravi Kiran Sarvadevabhatla. 2025. LineTR: Unified Text Line Segmentation for Challenging Palm Leaf Manuscripts. In *International Conference on Pattern Recognition*. Springer, 217–233.

[3] Yash Agrawal, Srinidhi Balasubramanian, Rahul Meena, Rohail Alam, Himanshu Malviya, and Rohini P. 2024. Optical Character Recognition using Convolutional Neural Networks for Ashokan Brahmi Inscriptions. *arXiv preprint arXiv:2501.01981* (2024). https://arxiv.org/abs/2501.01981

[4] G. Bhuvaneswari and G. Manikandan. 2019. Recognition of Ancient Stone Inscription Characters Using Histogram of Oriented Gradients. In *Proc. ICRTCCNT*. https://www.researchgate.net/publication/335308015_Recognition_of_Ancient_stone_Inscription_Characters_Using_Histogram_of_Oriented_Gradients

[5] Risab Biswas, Swalpa Kumar Roy, and Umapada Pal. 2023. A layer-wise tokens-to-token transformer network for improved historical document image enhancement. *arXiv preprint arXiv:2312.03946* (2023).

[6] Giordano Cicchetti and Danilo Comminiello. 2024. NAF-DPM: A Nonlinear Activation-Free Diffusion Probabilistic Model for Document Enhancement. arXiv:2404.05669 [cs.CV]

[7] Rachid Hedjam, Reza Farrahi Moghaddam, and Mohamed Cheriet. 2011. A spatially adaptive statistical method for the binarization of historical manuscripts and degraded document images. *Pattern Recognition* 44, 9 (2011), 2184–2196.

[8] Nicholas R. Howe. 2013. Document Binarization with Automatic Parameter Tuning. *Int. J. Document Analysis and Recognition* 16, 3 (2013), 247–258.

[9] Vishal K. et al. 2024. Review of Computational Epigraphy. *arXiv preprint arXiv:2406.06570* (2024). https://arxiv.org/abs/2406.06570

[10] Guisik Kim, Suhyeon Ha, and Junseok Kwon. 2018. Adaptive Patch Based Convolutional Neural Network for Robust Dehazing. In *2018 25th IEEE International Conference on Image Processing (ICIP)*. 2845–2849. https://doi.org/10.1109/ICIP.2018.8451252

[11] Ján Koloda and Jue Wang. 2023. Context aware document binarization and its application to information extraction from structured documents. In *International Conference on Document Analysis and Recognition*. Springer, 63–78.

[12] Haiping Lu, A.C. Kot, and Y.Q. Shi. 2004. Distance-reciprocal distortion measure for binary document images. *IEEE Signal Processing Letters* 11, 2 (2004), 228–231. https://doi.org/10.1109/LSP.2003.821748

[13] Monisha Munivel and VS Felix Enigo. 2024. MLIBT: A multi-level improvised binarization technique for Tamizhi inscriptions. *Expert Systems with Applications* 236 (2024), 121320.

[14] Mythic Society. 2024. Akshara Bhandara - Digital Stone Inscription Archive. https://mythicsociety.github.io/AksharaBhandara/#/. Accessed: 2025-06-24.

[15] Wayne Niblack. 1985. *An introduction to digital image processing*. Strandberg Publishing Company.

[16] Konstantinos Ntirogiannis, Basilis Gatos, and Ioannis Pratikakis. 2013. Performance Evaluation Methodology for Historical Document Image Binarization. *IEEE Transactions on Image Processing* 22, 2 (2013), 595–609. https://doi.org/10.1109/TIP.2012.2219550

[17] Ozan Oktay, Jo Schlemper, Loic Le Folgoc, Matthew Lee, Mattias Heinrich, Kazunari Misawa, Kensaku Mori, Steven McDonagh, Nils Y. Hammerla, Bernhard Kainz, Ben Glocker, and Daniel Rueckert. 2018. Attention U-Net: Learning Where to Look for the Pancreas. *arXiv preprint arXiv:1804.03999* (2018).

[18] Nobuyuki Otsu. 1979. A Threshold Selection Method from Gray-Level Histograms. *IEEE Transactions on Systems, Man, and Cybernetics* 9, 1 (1979), 62–66. https://doi.org/10.1109/TSMC.1979.4310076

[19] Ioannis Pratikakis, Konstantinos Zagoris, Xenofon Karagiannis, Lazaros Tsochatzidis, Tanmoy Mondal, and Isabelle Marthot-Santaniello. 2019. ICDAR 2019 Competition on Document Image Binarization (DIBCO 2019). In *2019 International Conference on Document Analysis and Recognition (ICDAR)*. 1547–1556. https://doi.org/10.1109/ICDAR.2019.00249

[20] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. 2015. U-Net: Convolutional Networks for Biomedical Image Segmentation. In *Proc. Medical Image Computing and Computer-Assisted Intervention (MICCAI) (LNCS, Vol. 9351)*. Springer, 234–241.

[21] Jaakko Sauvola and Matti Pietikäinen. 2000. Adaptive Document Image Binarization. *Pattern Recognition* 33, 2 (2000), 225–236.

[22] Giorgos Sfikas, Panagiotis Dimitrakopoulos, George Retsinas, Christophoros Nikou, and Pinelopi Kitsiou. 2024. Bessarion: Medieval Greek Inscriptions on a Challenging Dataset for Vision and NLP Tasks. In *International Workshop on Document Analysis Systems*. Springer, 393–407.

[23] Mohamed Ali Souibgui, Sanket Biswas, Sana Khamekhem Jemni, Yousri Kessentini, Alicia Fornés, Josep Lladós, and Umapada Pal. 2022. Docentr: An end-to-end document image enhancement transformer. In *2022 26th International Conference on Pattern Recognition (ICPR)*. IEEE, 1699–1705.

[24] Bolan Su, Shijian Lu, and Chew Lim Tan. 2013. Robust Document Image Binarization Technique for Degraded Document Images. *IEEE Trans. Image Processing* 22, 4 (2013), 1408–1417.

[25] Christopher Tensmeyer and Tony Martinez. 2017. Document Image Binarization with Fully Convolutional Neural Networks. In *Proc. 14th Int. Conf. on Document Analysis and Recognition (ICDAR)*. 448–453.

[26] The GIMP Development Team. 2024. GIMP - GNU Image Manipulation Program. https://www.gimp.org/. Accessed: 2025-06-24.

[27] Niharika Vadlamudi, Rahul Krishna, and Ravi Kiran Sarvadevabhatla. 2023. SeamFormer: High Precision Text Line Segmentation for Handwritten Documents. In *Proc. 17th Int. Conf. on Document Analysis and Recognition (ICDAR)*.

[28] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in neural information processing systems* 30 (2017).

[29] Wikimedia Commons Contributors. [n. d.]. Wikimedia Commons – Free Media Repository. https://commons.wikimedia.org/. Accessed: 2025-06-24.

[30] XP-Pen Technology Co., Ltd. 2024. XP-Pen Drawing Tablets for Digital Art and Annotation. https://www.xp-pen.com/. Accessed: 2025-06-24.

[31] Zongyuan Yang, Baolin Liu, Yongping Xiong, and Guibin Wu. 2024. GDB: gated convolutions-based document binarization. *Pattern Recognition* 146 (2024), 109989.

[32] Fernando-Luis Álvarez, Elena García-Barriocanal, and Joaquín-L. Gómez-Pantoja. 2010. Sharing Epigraphic Information as Linked Data. In *Metadata and Semantic Research*, Sánchez-Alonso and Athanasiadis (Eds.). Springer. https://doi.org/10.1007/978-3-642-16552-8_21