*Journal of Information Hiding and Multimedia Signal Processing* ©2026 ISSN 2073-4212
**Ubiquitous International** Volume **17**, Number **1**, March **2026**

# A Hybrid Vision Transformer-Based Pipeline for Structured Detection and Classification of Cuneiform Signs

Raed Majeed[1,*]

[1]Department of Computer Information Systems
University of Sumer
Dhi-Qar, Iraq
raed.m.muttasher@gmail.com

Ali Atshan Abdulredah[2]

[2]Department of Computer Science
University of Sumer
Dhi-Qar, Iraq
ali.atshan@uos.edu.iq

Hiyam Hatem[3]

[3]Department of Computer Science
University of Sumer
Dhi-Qar, Iraq
hiamhatem2005@gmail.com

[*]Corresponding author: Raed Majeed

Received July 25, 2025, revised October 3, 2025, accepted October 6, 2025.

Abstract. *Cuneiform is one of the earliest writing systems, consisting of wedge-shaped signs carved into clay tablets. These signs are structurally geometric and arranged in grid-like layouts, posing unique challenges for automated recognition. In this study, we propose a Vision Transformer (ViT)-based method for the detection and classification of cuneiform signs. Our approach integrates a robust pre-processing pipeline using Histogram Equalization, morphological operations, and Hough Transform to identify sign boundaries and segment individual signs from tablet images. To enhance structural feature learning, we apply skeletonization to each segmented sign, preserving topological properties such as wedge orientation and connectivity. Data augmentation is employed to simulate erosion, misalignment, and surface noise common in archaeological artifacts. Experiments demonstrate the effectiveness of our model in recognizing cuneiform signs with high accuracy, where performance metrics achieved accuracy of (0.89), precision of (0.93), recall of (0.90), and F1-score of (0.91), leveraging both raw and skeletonized representations. This work contributes toward the digitization and computational analysis of ancient cuneiform scripts. The presented method bridges the gap between classic line detection methods and modern deep learning-based classification.*

**Keywords:** Cuneiform Signs Classification, Convolutional Neural Networks (CNNs), Ancient Script Analysis, Vision Transformer, Skeleton Embedding.

1. **Introduction.** Cuneiform is one of the earliest known writing systems, developed in ancient Mesopotamia over 5000 years ago. Composed of wedge-shaped impressions on clay tablets, it was used to document administration, law, trade, and literature. The study of cuneiform is critical for understanding the culture, economy, and governance of early civilizations. However, manual transcription and interpretation are time-consuming and require extensive domain expertise [1]. The automatic recognition of cuneiform characters presents substantial challenges. These signs exhibit large intra-class variations due to tablet erosion, diverse scribal styles, and fragmentation. Furthermore, signs are typically densely packed, lack consistent alignment, and are carved in relief or impressed into non-uniform surfaces, making segmentation and classification difficult [2].
Earlier works in cuneiform recognition employed traditional image processing and feature extraction approaches, such as contour detection, template matching, and shape descriptors [3]. These techniques were often brittle and not scalable. In recent years, AI technologies have significantly advanced the fields of image analysis and computer vision in the development of highly accurate and efficient image recognition and classification models [4]. The rise of machine learning, particularly convolutional neural networks (CNNs), introduced a new wave of methods that offered improved robustness and accuracy [5]. Models such as YOLO and Faster R-CNN have been applied to detect and classify cuneiform signs at the object level [6]. More recently, transformer-based architectures like the Vision Transformer (ViT) have shown superior performance in capturing long-range spatial relationships in image data, outperforming CNNs in various image classification tasks [7]. Skeletonization techniques have also been leveraged to reduce the complexity of cuneiform characters by preserving their topological structure and minimizing visual noise [8] [9].
This study proposes a hybrid approach combining classical image processing with (ViT)-based classification to detect and classify Neo-Assyrian cuneiform signs. Our method includes pre-processing to enhance and segment sign boundaries, skeletonization to preserve structural geometry, and a Transformer-based classifier to improve recognition performance. The pipeline is validated using a labeled Neo-Assyrian group from (CLI) Dataset extracted from published epigraphic resources and evaluated through ablation studies and accuracy comparisons across multiple input representations. By automating the recognition of wedge-shaped signs in ancient cuneiform tablets, this work contributes toward scalable, efficient, and accurate digital epigraphy. Despite progress in the field, accurate cuneiform sign detection and classification remain limited by small datasets, unclear boundaries between signs, and the lack of structure-aware recognition techniques. There is a need for a robust pipeline that integrates both geometric priors and deep representation learning. figure (1) show the diversity of the cuneiform signs over several tablets.

FIGURE 1. Sample sets of cuneiform signs images and its corresponding representation on several tablets images.

The main contributions of our study can be summarized as:

- Apply Curriculum Learning to mimic the human learning process. Starting with the entire dataset, we first focus on simple signs, gradually introducing complex ones to enhance learning efficiency and performance.
- Introduce a novel pipeline that uses classical techniques for pre-processing and segmentation combined with (ViT) for classification.
- Apply skeletonization to preserve geometric and topological structure for improved recognition.
- Evaluate our approach using a CLI Dataset and conduct ablation studies comparing raw, skeleton, and hybrid inputs.

- Demonstrate that combining traditional and modern methods leads to more accurate and interpretable cuneiform recognition.

The remaining sections of this paper include: Section 2 reviews related works in cuneiform recognition. Section 3 outlines the proposed methodology and the pre-processing stage involved in the model. Section 4 presents experiments and results. Section 5 discusses findings and limitations. Section 6 concludes the study and proposes future directions.

2. **Related works.** The recognition of cuneiform signs using Vision Transformers (ViTs) is an emerging area of research that leverages deep learning techniques to address the complexities of ancient scripts. This approach enhances the accuracy of sign detection and classification, facilitating the study of historical artifacts. The following sections outline key Deep Learning Techniques. A Cuneiform Symbols Recognition System (CSRS) using the (VGG16) model presented to identify cuneiform symbols from the Code of Hammurabi; the model employed data augmentation and achieved high accuracy in recognizing intricate wedge-shaped symbols on clay tablets presented in [10]. A DL study in [3], specifically deep neural networks (DNNs), achieving an accuracy of (83%) in recognizing cuneiform symbols. This approach demonstrated improved performance compared to traditional machine learning algorithms when applied to a balanced dataset using unigram features. However, these are some limitation relating to lack of information on cuneiform languages and imbalanced categories in the (CLI) dataset. (SVM, KNN, DT, RF) achieved accuracies of (88.15%, 88.14%, 94.13%, 95.46%).
A DL-based cuneiform sign detector that utilizes weak supervision from existing transliterations to localize and classify cuneiform signs in images, enabling efficient training without extensive manual annotations, thus Transliteration lacks visual detail from tablet images and the rare sign code classes are underrepresented in training data [11]. The study introduced a large dataset for cuneiform sign detection and Evaluated performance with manually annotated bounding boxes. [12] Employs a DL-based human-in-the-loop (OCR) pipeline specifically designed for recognizing cuneiform symbols in transliterated texts. It achieves a character error rate of (9%) on clean data, enhancing accessibility for computational analysis of these ancient documents. The (OCR) systems are insufficient for digitization. DeepScribe employs a modular computer vision pipeline using a RetinaNet object detector for localization and a ResNet classifier for sign identification, achieving a localization (mAP) of (0.78) and a top-5 classification accuracy of (0.89) for cuneiform symbols [13].
A DL approach using a Transformer network for Cuneiform Language Identification in [14], achieving (77%) accuracy on test data, which established a new state-of-the-art performance in the CLI evaluation. A study in [15] employs multilayer neural networks for cuneiform symbol recognition, utilizing K-means clustering to group similar symbols. Features extracted include vertical and horizontal projections, center of gravity, and connected components, enhancing recognition rates effectively. The study main Limitation can be summarized in two points: the use of certain strokes in cuneiform symbols and rare occurrences of inverse vertical strokes on old tablets. A DL approach for cuneiform symbol recognition using illumination-based data augmentation presented in [16], enhancing classification accuracy by generating synthetic training data from 3D datasets, thus overcoming challenges related to illumination variations in photographic reproductions. The approach main limitation are Poor transferability due to different lighting setups and image artifacts and insufficient illumination information from certain training datasets, achieving(90%)accuracy with a (28.5%) reduction in classification error through illumination augmentation.
While the advancements in deep learning for cuneiform recognition are promising, challenges remain, particularly in the need for high-quality annotated datasets and the complexity of the script itself, which may hinder broader applications in the field.

3. **The Methodology.** Our approach to cuneiform sign recognition integrates classical image processing with modern deep learning, enabling both accurate sign segmentation and classification. The pipeline consists of multiple stages. Figure (2) illustrate the model architecture.

3.1. **Pre-processing and Edge Detection.** Cuneiform tablets are often degraded due to age, erosion, or uneven surfaces. To prepare images for segmentation, we apply Grayscale Conversion to obtain the dataset images to grayscale mode, then we standardize the pixel intensity values by normalization step, Contrast Limited Adaptive Histogram Equalization (CLAHE) to enhance local contrast, followed by Gaussian blurring to reduce noise.

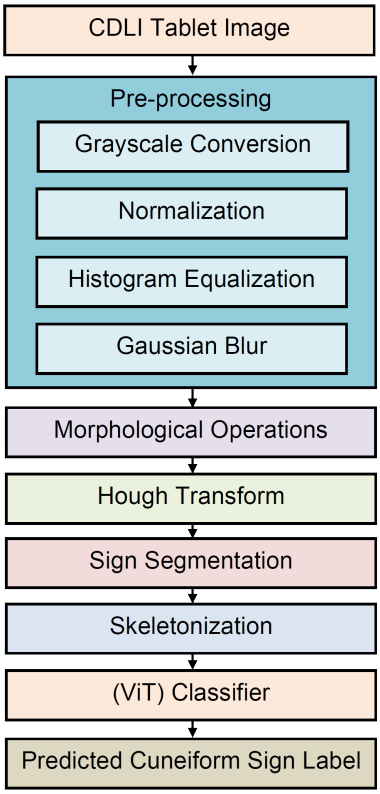


FIGURE 2. The model block diagram.

3.2. **Grayscale Conversion.** Cuneiform signs are shape-based not color-based so all the extracted features depend on the shape of the cuneiform signs, it simplifies the model input, Reduces overfitting on background color/noise and emphasizes the structural patterns of the signs engraved in tablets image. Therefore, all RGB images were converted to grayscale using the standard luminance transformation:

$$I = 0.229.R + 0.587.G + 0.114B \tag{1}$$

3.3. **Normalization.** To standardize the pixel intensity values, each image was normalized to the range [0,1] by dividing the pixel values by 255, this normalization facilitates stable training dynamics by reducing numerical instability in gradient computations.

$$I_{\text{norm}} = \frac{1}{255} \tag{2}$$

3.4. **Contrast Limited Adaptive Histogram Equalization (CLAHE).** CLAHE enhances local contrast by applying histogram equalization over small contextual regions (tiles) and limiting contrast amplification to avoid noise enhancement. Let the input grayscale image be denoted by $I(x, y)$, where $(x, y)$ represents the pixel coordinates. Let:

- $T_{i,j}$ be a local tile (sub-region) of the image,
- $H_{T_{i,j}}$ be the histogram of tile $T_{i,j}$,
- $C$ be the clip limit threshold to prevent over-amplification,
- $\mathcal{I}^{-1}$ be the inverse mapping that reconstructs intensity values from the histogram.

The contrast-limited histogram is defined as:

$$H'_{T_{i,j}}(k) = \min(H_{T_{i,j}}(k), C) \tag{3}$$

The (CLAHE)-enhanced image is obtained by:

$$I_{\text{CLAHE}}(x, y) = \mathcal{I}^{-1}(H'_{T_{i,j}}) \tag{4}$$

Bilinear interpolation is applied between neighboring tiles to ensure smooth transitions across tile boundaries.

3.5. **Gaussian Blurring.** Gaussian blurring reduces high-frequency noise by convolving the image with a Gaussian kernel. The blurred image $I_G(x, y)$ is computed as:

$$I_G(x, y) = \sum_{i=-k}^{k} \sum_{j=-k}^{k} G(i, j) \cdot I_{\text{CLAHE}}(x - i, y - j) \tag{5}$$

where the Gaussian kernel $G(i, j)$ is defined as:

$$G(i, j) = \frac{1}{2\pi\sigma^2} \exp\left(-\frac{i^2 + j^2}{2\sigma^2}\right) \tag{6}$$

Here:

- $\sigma$ is the standard deviation of the Gaussian kernel,
- $k$ defines the kernel radius (typically $k = 3\sigma$).

3.6. **Morphological Operations.** To refine the edge map and close gaps in sign contours, we use morphological dilation with a small rectangular kernel. This helps to consolidate fragmented wedges into connected regions; a morphological opening can be applied to remove small noise artifacts without affecting the core structure.

3.6.1. *Dilation.* Dilation expands the foreground (white) regions in a binary image, helping to connect fragmented edges. Given a binary image $B(x, y)$ and a structuring element $S$ (typically a small rectangular kernel), the dilation operation is defined as:

$$B_{\text{dilated}}(x, y) = \max_{(i,j) \in S} B(x - i, y - j) \tag{7}$$

This operation ensures that any pixel in the neighborhood defined by $S$ that is 'on' (1) causes the output pixel to also be 'on'.

3.6.2. *Opening.* Opening removes small noise while preserving the overall structure of the objects. It is defined as erosion followed by dilation:

$$B_{\text{opened}} = (B \ominus S) \oplus S \tag{8}$$

where: - $\ominus$ denotes (erosion): shrinking the foreground by removing boundary pixels, - $\oplus$ denotes (dilation), as defined above. Opening is especially useful in eliminating small isolated regions (noise) without affecting the larger wedge structures of cuneiform signs.

3.7. **Hough Transform.** Cuneiform signs are typically arranged in a structured grid-like layout of horizontal rows and vertical columns. To leverage this spatial organization, we apply the Hough Line Transform to detect the dominant linear structures in the image, enabling the extraction of candidate bounding boxes for signs and for ensuring consistent alignment across rows and columns. The Hough Transform maps edge points from the image domain $(x, y)$ into a parameter space $(\rho, \theta)$ using the following relationship:

$$\rho = x \cos\theta + y \sin\theta \tag{9}$$

where:

- $\rho$ is the perpendicular distance from the origin to the line,
- $\theta$ is the angle between the x-axis and the normal vector to the line.

An accumulator array is used to collect votes for potential line parameters $(\rho, \theta)$ associated with each edge point. Peaks in the accumulator indicate the most prominent lines in the image.

To detect the grid structure of cuneiform text:

- Horizontal cuneiform signs lines are detected by focusing on angles $\theta = 0°$.
- Vertical cuneiform signs lines are detected by focusing on angles $\theta = 90°$.

The intersections of horizontal and vertical lines define rectangular grid cells, which are used as (candidate bounding boxes) for individual signs. This strategy ensures spatial consistency in sign detection across rows and columns, especially in densely packed or degraded tablet images.

3.8. **Sign Segmentation and Cropping.** Based on the detected lines and contours, we extract individual sign regions using bounding boxes. We filter out noise by size thresholds and preserve only regions within a plausible sign dimension range. To ensure consistent input dimensions across the dataset, all signs images were resized to a fixed dimensions suitable for the object detection model (224 × 224) pixels. This resizing allows the model to batch process sign images efficiently and match the input size expected by the detection architecture. The resizing operation was carried out using bilinear interpolation to preserve edge features of the wedge-shaped symbols. Resizing maps each output pixel coordinate (x',y') to its corresponding position in the original cropped sign image (x,y) as follows:

$$x = \frac{x'}{W_{\text{out}}} \cdot W_{\text{in}}, \quad y = \frac{y'}{H_{\text{out}}} \cdot H_{\text{in}} \tag{10}$$

The intersections define candidate bounding boxes for individual cuneiform signs. These regions are extracted from the original image using bounding box coordinates. To reduce false positives and eliminate irrelevant noise, we apply size-based filtering. Let each bounding box $B_i$ have width $w_i$ and height $h_i$. We retain only those boxes satisfying:

$$w_{\min} \leq w_i \leq w_{\max}, \quad h_{\min} \leq h_i \leq h_{\max} \tag{11}$$

where $[w_{\min}, w_{\max}]$ and $[h_{\min}, h_{\max}]$ define acceptable size ranges for plausible cuneiform signs, empirically determined based on dataset characteristics. Each valid region is then cropped and resized to a standard resolution of 224 × 224 pixels using bilinear interpolation. This fixed input size ensures compatibility with downstream (ViT) classifiers while preserving essential visual features for recognition. Figure (3) present the Detection of horizontal and vertical lines on a cuneiform tablet using the Hough Transform. Line intersections define the grid for sign segmentation.

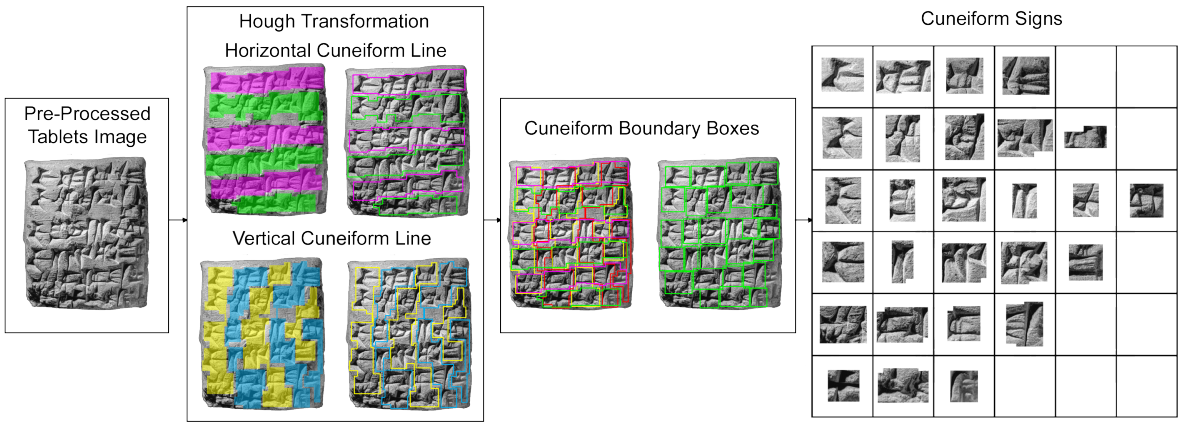


FIGURE 3. Hough Transform sign segmentation.

3.9. **Skeletonization.** Cuneiform signs are composed of geometric wedge-shaped strokes arranged in structured spatial patterns. These signs were inscribed into clay tablets and exhibit consistent structural rules based on orientation, size, and placement. Although cuneiform does not possess artistic stroke variation like cursive calligraphy, its historical and geometric complexity still poses challenges for automated recognition. Importantly, accurate identification can often be achieved using structural features alone, without requiring texture or intensity cues. Examples of the extracted skeletons are shown in Figure (4).

4. **The Models.** We employed the base models: (TrOCR) [17] and (ViTSTR) [18]. (TrOCR) uses the standard Transformer Encoder-Decoder structure. The (TrOCR model) contains two parts: an image transformer to capture features of images and a sings transformer to generate cuneiform piece sequences. (ViTSTR) is a Vision Transformer-based model that, compared to ViT, can recognize multiple sign characters while ensuring the correct sequence and length of the sign symbol. The only difference between (ViT) and (ViTSTR) is the prediction head, which is used to recognize multiple signs and their sequence. To enhance Vision Transformer (ViT) models for the task of cuneiform sign classification, we draw inspiration from existing research in skeletonization-based recognition tasks [19]. While much of the

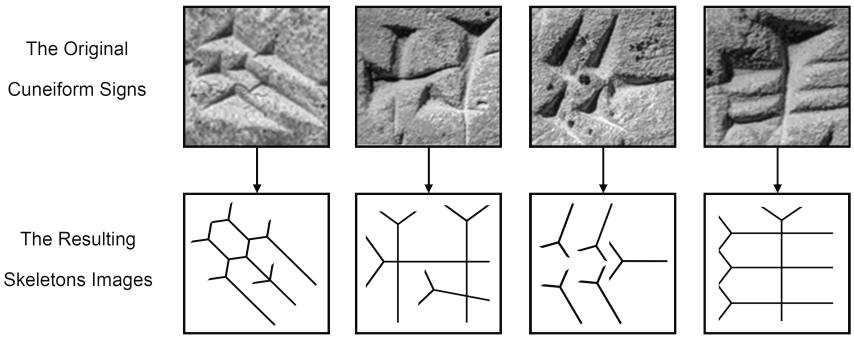


FIGURE 4. Sample extracted cuneiform signs from the CLI dataset and the corresponding one pixel wide skeletons.

prior work in this area has focused on ideographic scripts [20], the underlying principles of topological simplification also apply to wedge-based scripts like cuneiform. In our approach, we apply skeleton extraction to each segmented cuneiform sign image in the dataset. This involves iteratively removing edge pixels while preserving a single-pixel-wide skeleton that retains the sign's geometric and topological structure, including wedge orientation and connectivity. The resulting skeletons serve as a simplified structural representation of the original signs, providing a form of prior knowledge to support training and improve classification accuracy.

4.1. **Model Notation.** The notations used in this section are presented in Table (1).

TABLE 1. Notation used in this section.

| Symbol | Description |
|---|---|
| $i$ | A training sample |
| $x_i$ | An input feature of training sample $i$ |
| $L(x_i)$ | Cross-entropy loss for the sample $i$ |
| $C(x_i)$ | Confidence predicted for the sample $i$ |
| $D(i)$ | Difficulty metric for the sample $i$ |
| $\tau$ | Threshold for classifying a sample as easy or hard |
| $\alpha$ | Weight coefficient for the cross-entropy loss |
| $\beta$ | Weight coefficient for the confidence |
| $ES$ | Set of easy samples |
| $HS$ | Set of hard samples |
| $L_{\text{train}}$ | The training loss |
| $T_{\text{pre}}$ | Duration of the pretraining phase |
| $T_{\text{CL}}$ | End time of the dynamic curriculum learning phase |
| $N$ | Total number of samples in the training set |
| $N_{\text{easy}}$ | Number of easy samples |
| $p_j$ | The $j$-th flattened image patch |
| $p_{\text{class}}$ | The class token |
| $s_j$ | The $j$-th flattened skeleton patch |
| $E$ | Learnable embedding matrix for image patches |
| $E_{\text{pos}}$ | Positional embedding matrix |
| $E_s$ | Learnable embedding matrix for skeleton patches |
| $\|$ | Concatenation operation along the embedding dimension |
| $z_0$ | Combined embedding as input of ViT encoder |

4.2. **Training with Curriculum Learning.** The concept of curriculum learning, derived from the educational system in human society [21], begins with easy concepts and gradually introduces more difficult ones to improve the efficiency and effectiveness of learning. Typically, we start by filtering out the hard samples from the training set, focusing on a subset of easier samples for the initial stages of

model training. After a certain number of training epochs, we gradually introduce subsets containing the more challenging samples. The two key challenges in this stage arise:

- Distinguishing between easy and hard samples.
- Deciding when to incorporate the harder subsets into the training process.

These methods are predefined, relying on prior human knowledge, such as manually annotated data sets, task-specific complexity metrics, and the popular scheduler Baby Step [22], which groups training data by difficulty and merges them after certain epochs. However, most current cuneiform scripts have no tag complexity metric for indicating the classification difficulty. Thus, we choose to deal with this problem during training process. To address these challenges, researchers decompose CL into two independent yet closely related subtasks [23], which is also the strategy we applied in our experiments. In this paper [24], these two subtasks are abstracted as Difficulty Measurer and Training Scheduler.

4.3. **Difficulty Measure.** For the input feature ($x_i$) of each training sample i, the cross-entropy loss $L(x_i)$ and confidence C ($x_i$) are computed by the classification model. The difficulty $D(i)$ of each sample is then assessed using a combined metric.

$$D(i) = \alpha \cdot L(x_i) + \beta \cdot (1 - C(x_i)) \tag{12}$$

Where $\alpha$ and $\beta$ adjust the weights of the loss and confidence. The sample is classified as easy or hard based on whether $D(i)$ exceeds a threshold.

$$\text{Difficulty}(i) = \begin{cases} \text{Hard Sample (HS)}, & \text{if } D(i) > \tau \\ \text{Easy Sample (ES)}, & \text{if } D(i) \leq \tau \end{cases} \tag{13}$$

4.4. **Training Scheduler.** Initially, the entire training dataset pre-processed images is used for model training, which denoted as the pre-training phase $T_{\text{pre}}$ . The training loss $L_{\text{train}}$ at this stage includes all samples

$$L_{\text{train}} = \frac{1}{N} \sum_{i=1}^{N} L(x_i), \quad \text{for } 0 \leq t \leq T_{\text{pre}} \tag{14}$$

During the dynamic curriculum learning phase, from $T_{\text{pre}}+1$ to $T_{\text{pre}}+n$, the training loss only includes easy samples:

$$L_{\text{train}} = \frac{1}{N_{\text{easy}}} \sum_{i \in \text{ES}}^{N} L(x_i), \quad \text{for } T_{\text{pre}} + 1 \leq t \leq T_{\text{pre}} + n \tag{15}$$

After this phase, from $t > T_{\text{pre}} + n$, both easy and hard samples are used:

$$L_{\text{train}} = \frac{1}{N} \left( \sum_{i \in \text{HS}} L(x_i) + \sum_{i \in \text{ES}} L(x_i) \right), \quad \text{for } t > T_{\text{pre}} + n \tag{16}$$

4.5. **ViT with Skeleton Embedding.** Vision Transformer (ViT) [7] decomposes each image into a sequence of tokens with a fixed length, where the tokens represent non-overlapping image patches. The cuneiform image from Neo-Assyrian signs lists are converted to the skeleton image.

$$z_0 = [p_{\text{class}};\, p_1 E;\, p_2 E;\, \ldots;\, p_N E] + E_{\text{pos}}$$

where $z_0$ is the input of ViT encoder, $p_{\text{class}}$ is the class token, and $E_{\text{pos}}$ is the positional embedding matrix.

In our modified ViT architecture, we integrate skeleton embeddings extracted from the images. For each sample, the embeddings from the image and its corresponding skeleton are concatenated before being fed into the ViT encoder. The combined embedding sequence is:

$$z_0 = [p_{\text{class}};\; p_1 E \,\|\, s_1 E_s;\; p_2 E \,\|\, s_2 E_s;\; \ldots;\; p_N E \,\|\, s_N E_s] + E_{\text{pos}} \tag{17}$$

where $p_j E$ and $s_j E_s$ represent the embeddings of the image and skeleton patches, respectively. The process is illustrated in Figure(5).

## 5. **Experimentation.**

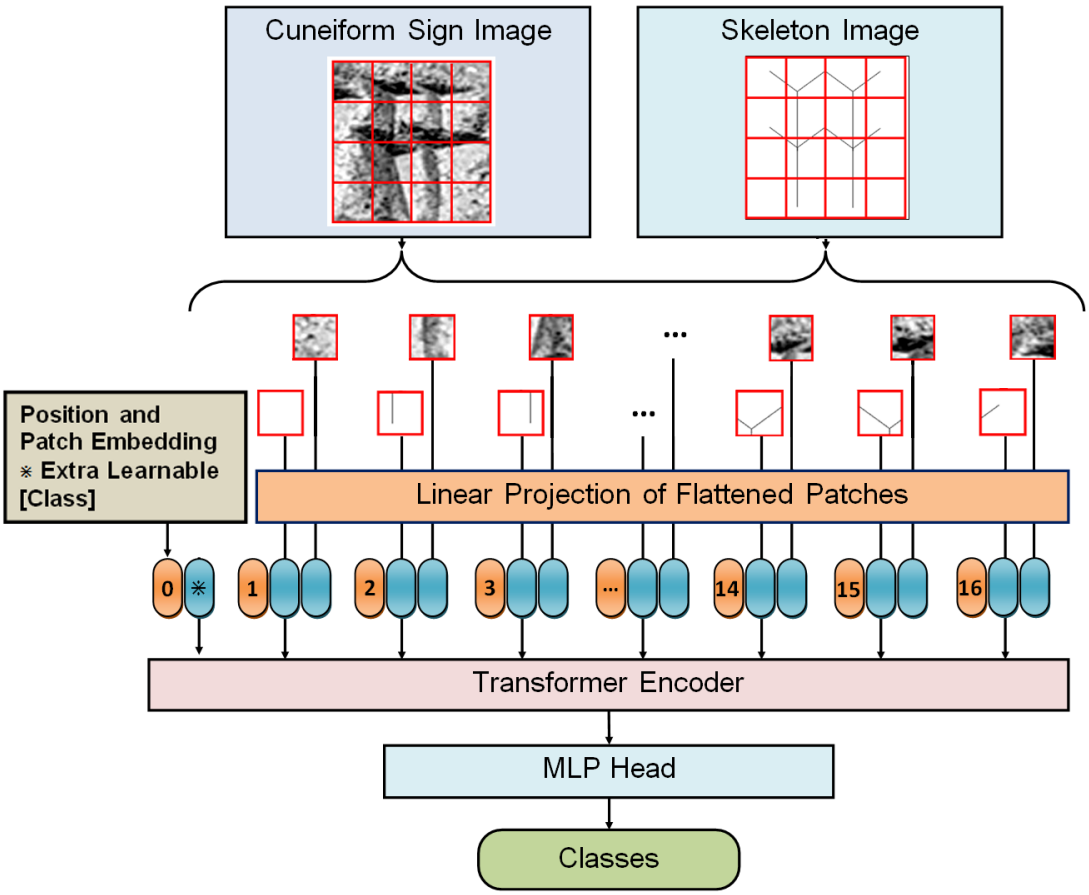


FIGURE 5. (ViT) architecture with skeleton embedding. The input image and skeleton embeddings are concatenated and combined with positional encoding before being fed into the encoder.

5.1. **The Dataset.** The CLI dataset comprises 134,000 portions of cuneiform texts associated with seven dialects of languages written in cuneiform,. Provide a concise overview of the classifications for the various categories of cuneiform classes' images [25]. The figure illustrates that the classes SUX and NEA encompass the majority of the data relative to other classes, highlighting the imbalance presenting among classes in the dataset. The CLI dataset exhibits an imbalance in class distribution; this phase executes the up-sampling of minority classes within the CLI dataset. New samples are generated by randomly selecting samples from the minority classes with replacements. The original dataset has been supplemented with new samples to achieve a balanced distribution, resulting in 53,673 samples per class, as illustrated in Table (2).

TABLE 2. CLI Dataset classes after the Balancing.

| No. | Class | No. of Samples | Samples After Balance | Percentage |
|---|---|---|---|---|
| 1 | LTB | 15,947 | 53,673 | 29.7% |
| 2 | MPB | 5,508 | 53,673 | 10.3% |
| 3 | NEA | 32,966 | 53,673 | 61.5% |
| 4 | NEB | 9,707 | 53,673 | 18.1% |
| 5 | OLB | 3,803 | 53,673 | 7.1% |
| 6 | STB | 17,817 | 53,673 | 33.2% |
| 7 | SUX | 53,673 | 53,673 | 100% |

5.2. **Dataset Splitting.** The dataset is divided into three primary sections:

1. The training set constitutes the largest portion of the dataset, accounting for 80%, and serves the purpose of training the proposed models while adjusting the weights through the observation and learning of the correct output.
2. The validation set represents a portion of the dataset, specifically (10%). This component is utilized to assess the model through the adjustment of hyper parameters. This data exerts an indirect influence on the models, as it is observed by them but not employed for learning objectives.
3. The testing set (10%) serves as a distinct part of the dataset, employed to ensure an impartial and precise assessment of the models following the conclusion of the entire training process.

Our experiments are based on a labeled dataset of Neo-Assyrian cuneiform group from high-resolution images of clay tablets. The dataset contains a total of 134,000 individual sign images across 907 unique sign classes. Each image corresponds to a segmented sign extracted from a tablet using our pre-processing and Hough-based detection pipeline. To ensure a diverse and representative dataset, signs were collected from multiple tablets with varying degrees of surface erosion and inscription clarity.

5.3. **Data Augmentation.** Given the structural and geometric consistency of cuneiform signs composed primarily of wedge-shaped elements, we adopt an augmentation strategy inspired by contrastive learning to support the training of robust and discriminative representations [26] [27], cuneiform signs preserve distinct geometric boundaries, making augmentation essential for addressing surface noise, illumination variation, and carving inconsistency in ancient tablets. To improve the generalization ability of our ViT-based model for cuneiform sign recognition, we apply the following augmentation techniques to both the raw and skeletonized sign images:

- Random Translation in both horizontal and vertical directions by a factor between 10% and 20% of the image dimensions. This mimics variability in sign placement due to erosion or tablet tilt.
- Gaussian Blur using a kernel size randomly selected between 11 and 21, simulating de-focus or clay surface smoothness.
- Gaussian Noise with a standard deviation randomly chosen between 0.3 and 0.5, to model sensor noise or lighting inconsistencies in the dataset images.

These transformations are applied in a fixed sequence per image to simulate natural variance found in cuneiform tablets while preserving the core wedge structure. Both original and augmented images are retained, effectively expanding the training dataset and improving the model's robustness to style, noise, and degradation. This is particularly beneficial when paired with the skeletonization process, as the model learns from both detailed visual inputs and abstracted structural forms.

5.4. **Experimentation and Discussion.** For model implementation we use 4 NVIDIA GeForce RTX 3090 GPUs with the memory of 24GBs for training and 1 NVIDIA GeForce RTX 3090 GPU for testing. The (ViT) model is trained on both raw and skeletonized sign images, resized to 224x224 pixels. We use a cross-entropy loss function and optimize the model using Adam with a learning rate of $1 \times 10^{-4}$. Training is performed over 120 epochs with an 80/10 train-test split.

For the initial 256 iterations, the dataset is dedicated entirely to the training process without exception. As the training progresses from batch 257 to batch 1000, a loss-based filtering criterion is introduced, where batches are discarded if their loss exceeds a threshold $\tau$ of 3. Subsequently, from batch 1001 up until the 2000th batch, this threshold $\tau$ is lowered to 1, tightening the criteria for batch inclusion. Beyond the 2000th batch, the threshold $\tau$ is further reduced to 0.1, allowing for a more inclusive approach where the vast majority of samples contribute to the training phase.

During the training process, the confidence score will continue to rise. We calculate each sample's confidence score, comparing it to the recent average confidence score, which resets every 1,024 samples. Samples with scores below average are masked, and if the number of masked samples in a batch exceeds the threshold, the batch will be skipped. The masking threshold also adjusts during training based on the recent average masked rate.

When computing the difficulty $D(i)$ of samples i, there are two cases of the value of $\alpha$ and $\beta$, which are the weight coefficients of cross-entropy loss and confidence scores,

- $\alpha = 1$, $\beta = 0$
- $\alpha = 0$, $\beta = 1$

We evaluate model performance using accuracy, precision, recall, and F1-score. These metrics are computed across all classes to assess the model's ability to distinguish between visually similar signs. Additional qualitative results are presented to show predictions on challenging examples, including signs with erosion, partial wedges, or noise artifacts.

6. **Results and Model Comparisons.** Our (ViT)-based model achieved strong performance on the cuneiform sign dataset. By leveraging both raw and skeletonized inputs, the model attained a classification accuracy of 89.60, with macro-averaged precision of 0.93, recall of 0.90, and F1-score of 0.915. The model also reached an AUC score of 0.94, indicating high discriminative capability across sign classes. These results confirm the effectiveness of integrating visual texture and topological structure for robust sign recognition. The inclusion of skeletonized images significantly improved model performance, especially when used alongside raw image inputs. This improvement was most evident when the batch size was set to 32. In contrast, doubling the batch size to 64 yielded marginal gains, suggesting diminishing returns. The structural cues preserved in skeletons proved critical for differentiating signs with similar textures or those affected by erosion. Moreover, models trained solely on skeleton images demonstrated competitive performance, highlighting the standalone value of structural information for classification tasks. However, combining both representations consistently produced the best results. figure (6) show the model confusion matrix.

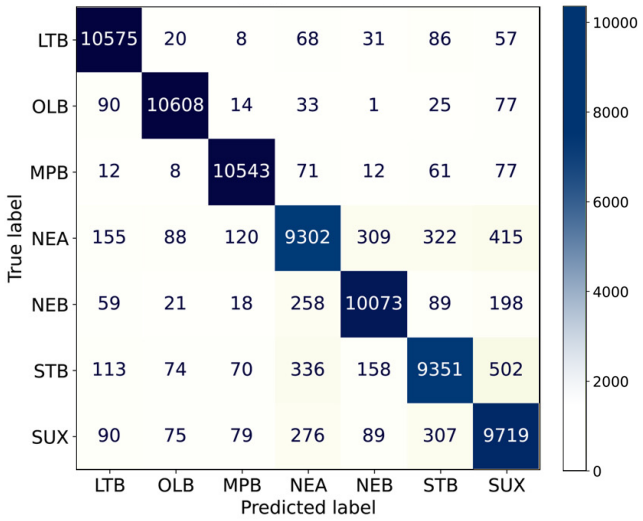


FIGURE 6. The confusion matrix performance.

The pre-processing pipeline, including CLAHE, Canny edge detection, and morphological operations, played a crucial role in enhancing sign segmentation quality. It enabled more accurate bounding box generation, reduced background noise, and facilitated cleaner skeleton extraction. Without this preprocessing stage, the model exhibited a measurable drop in accuracy due to inconsistent or noisy input data. table (3) show the model performance evaluation metrics values.

TABLE 3. The performance evaluation results.

| Experiment | Accuracy | Precision | Recall | F1-Score |
|---|---|---|---|---|
| Full Model | 89.60 | 0.93 | 0.91 | 0.92 |
| Skeletonization | 87.80 | 0.91 | 0.89 | 0.90 |
| Data Augmentation | 88.10 | 0.91 | 0.90 | 0.91 |

The (ViT) model achieved its optimal performance with a batch size of 32, whereas the TrOCR model performed best with a batch size of 8. Notably, incorporating ResNet as a feature extractor reduced the classification accuracy of the (ViT) model, likely due to redundancy between convolutional and attention-based representations. Curriculum learning further boosted the (ViT) model's performance. At a batch size of 32, both loss-based and confidence-based curriculum strategies led to significant accuracy improvements. However, at a batch size of 64, only the loss-based CL maintained performance gains, while confidence-based CL slightly degraded results possibly due to overfitting or reduced sample diversity per batch. For the (TrOCR) model, curriculum learning using either loss functions or confidence scores had minimal impact on recognition performance, suggesting that its pretrained encoder-decoder architecture

already captures robust representations without the need for progressive learning strategies.figure (7) show the Validation/training accuracy and loss for both (VIT) model and (TrOCR) Model and table (4) present the performance comparison with other related works.

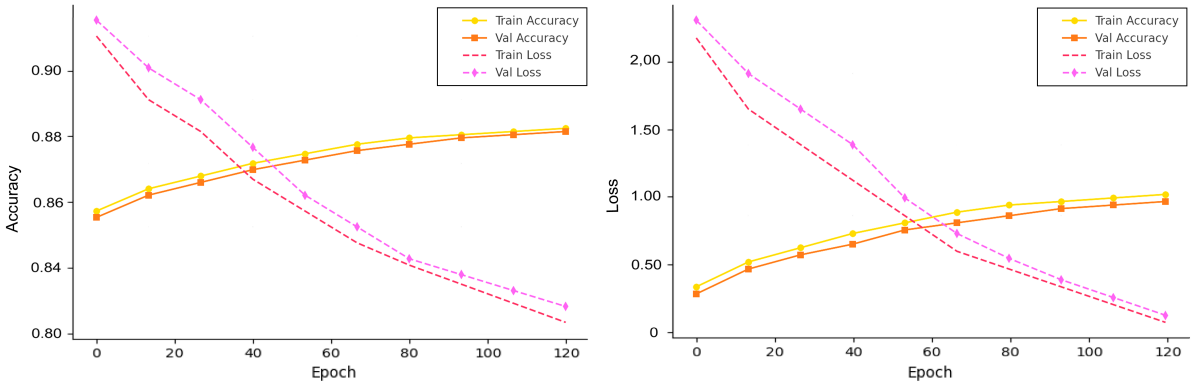


FIGURE 7. The proposed (VIT) model Validation/training accuracy and loss using CLI dataset for (a) The proposed (VIT) model and (b) TrOCR Model.

TABLE 4. Comparison with related approaches.

| Reference | Method | Dataset | Performance Metrics |
|---|---|---|---|
| [28] (2024) | CNN | Custom dataset for stylistic variants | Accuracy: 83% |
| [16] (2022) | CNN | 3D scans projected to 2D | Accuracy: 80% |
| [13] (2025) | Modified CNN | Elamite tablets (UChicago dataset) | Accuracy: 82% |
| [11] (2020) | CNN | Cuneiform 2D image dataset | Accuracy: 81% |
| [29] (2024) | ResNet50 and VAE | CDLI (94,000 images) | F1-Score: 61% |
| Proposed Model | ViT + ResNet | CLI (balanced 2D images) | Accuracy: 89.60% |

6.1. **Ablation Study.** To assess the contribution of each pipeline component, we conduct an ablation study by comparing model performance with and without skeletonization, augmentation, and Hough-based segmentation. Results confirm that the integration of structural pre-processing and skeleton embedding significantly improves recognition accuracy and robustness. To evaluate the effectiveness of each component in our proposed recognition pipeline, we conduct an ablation study using two batch sizes: 32 and 64. We systematically vary the presence of key components: Skeleton Embedding (Sklt), Curriculum Learning (CL), and Feature Extraction (FE) using a pre-trained ResNet. Removing skeletonization resulted in a decrease of approximately 2.0% in F1-score, while omitting data augmentation led to a 1.5% drop. Eliminating Hough-based segmentation had the most severe impact, reducing classification accuracy by 4.3%, highlighting its essential role in producing well-aligned and non-overlapping sign crops critical for model performance figure (8) show the performance drop over stage modification . Interestingly, for batch size 64, performance trends are mostly consistent, with a slight drop in accuracy but better stability in precision across configurations. This suggests that moderate batch sizes (32) may provide better generalization when training on smaller or noisier datasets like cuneiform symbols. Overall, the ablation study confirms that curriculum learning and skeleton embedding each contribute positively to model performance. However, combining all components does not always yield additive benefits, highlighting the importance of carefully balancing complexity and representation diversity in historical script recognition.

6.2. **Study Limitations.** While the proposed method performs well on high-quality tablet images, however, there are some limitations related to factors like: Poor qualities in training datasets have an impact on the model production, some of the clay tablets images are eroded and a large proportion of the textual contents are missing. Cuneiform image vary in condition and dimensions, cutting and part missing can affect model achievements. While small size image required more enhancement processes. The

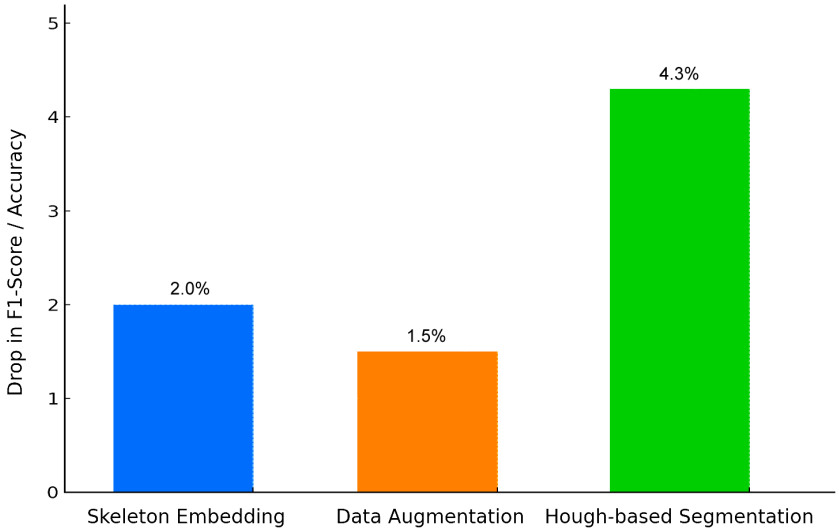


FIGURE 8. The performance drop when removing model components.

diverse system of writing for the (cuneiform signs) across time periods such as; (Sumerian, Akkadian, etc.) represent a limitation for future translation. The segmentation stage in some case obtains boundary box for overlapped sings which require iteration in hough transform procedure to achieve separated cuneiform sings. Overall, our results confirm that combining classical image processing with Transformer based deep learning creates a strong framework for accurate and scalable cuneiform sign recognition.

7. **Conclusion.** In this work, we presented a comprehensive pipeline for the detection and classification of cuneiform signs using a Vision Transformer (ViT) model. Our method combines classical image pre-processing techniques including; morphological operations, and Hough Transform with modern deep learning. We introduced a skeletonization step to preserve the geometric structure of wedge-based signs and enhance classification accuracy. Experimental results demonstrate that our approach is effective in recognizing cuneiform signs under challenging conditions, including erosion, noise, and structural overlap. The inclusion of both raw and skeletonized images, coupled with contrastive-inspired augmentation strategies, significantly improved model robustness. Future work may explore integrating linguistic context and temporal sequence modeling to interpret complete cuneiform inscriptions, as well as the application of this pipeline to 3D tablet scans. Our results pave the way for scalable and accurate digitization of ancient scripts and support broader applications in digital epigraphy and Assyriology.

## REFERENCES

[1] H. Hameeuw, K. De Graef, G. R. Smidt, A. Goddeeris, T. Homburg, and K. Kumar Thirukokaranam Chandrasekar, "Preparing multi-layered visualisations of old babylonian cuneiform tablets for a machine learning ocr training model towards automated sign recognition," *it-Information Technology*, vol. 65, no. 6, pp. 229–242, 2024.

[2] E. A. Saeed, A. D. Jasim, and M. A. A. Malik, "Cuneiform text dialect identification using machine learning algorithms and natural language processing (nlp)," *Iraqi Journal of Information and Communication Technology*, vol. 7, no. 2, pp. 26–40, 2024.

[3] M. Mahmood, F. M. Jasem, A. A. Mukhlif, and B. AL-Khateeb, "Classifying cuneiform symbols using machine learning algorithms with unigram features on a balanced dataset," *Journal of Intelligent Systems*, vol. 32, no. 1, p. 20230087, 2023.

[4] R. Majeed, H. Hatem, and M. Kaisb, "Develop oil and gas production forecasting models based on deep learning and machine learning," *Journal of Information Hiding and Multimedia Signal Processing*, vol. 16, no. 3, pp. 966–983, 2025.

[5] E. Stötzner, T. Homburg, and H. Mara, "Cnn based cuneiform sign detection learned from annotated 3d renderings and mapped photographs with illumination augmentation," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023, pp. 1680–1688.

[6] E. A. Saeed, A. D. Jasim, and M. A. A. Malik, "Deciphering the past: enhancing assyrian cuneiform recognition with yolov8 object detection," *International Journal of Advanced Technology and Engineering Exploration*, vol. 10, no. 109, p. 1604, 2023.

[7] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly *et al.*, "An image is worth 16x16 words: Transformers for image recognition at scale," *arXiv preprint arXiv:2010.11929*, 2020.

[8] T.-Q. Wang and C.-L. Liu, "Fully convolutional network based skeletonization for handwritten chinese characters," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 32, no. 1, 2018.

[9] M. Mahalakshmi and M. Sharavanan, "Ancient tamil script and recognition and translation using labview," in *2013 International conference on communication and signal processing*. IEEE, 2013, pp. 1021–1026.

[10] S. Elshehaby, M. Al-Saad, A. Panthakkan, and H. Al Ahmad, "Unlocking ancient secrets: A deep learning approach to cuneiform symbols recognition," in *2024 Advances in Science and Engineering Technology International Conferences (ASET)*. IEEE, 2024, pp. 1–6.

[11] T. Dencker, P. Klinkisch, S. M. Maul, and B. Ommer, "Deep learning of cuneiform sign detection with weak supervision using transliteration alignment," *Plos one*, vol. 15, no. 12, p. e0243039, 2020.

[12] S. Gordin, M. Alper, A. Romach, L. S. Santos, N. Yochai, and R. Lalazar, "Cured: Deep learning optical character recognition for cuneiform text editions and legacy materials," in *Proceedings of the 1st Workshop on Machine Learning for Ancient Languages (ML4AL 2024)*, 2024, pp. 130–140.

[13] E. C. Williams, G. Su, S. R. Schloen, M. Prosser, S. Paulus, and S. Krishnan, "Deepscribe: localization and classification of elamite cuneiform signs via deep learning," *ACM Journal on Computing and Cultural Heritage*, vol. 18, no. 2, pp. 1–32, 2025.

[14] G. Bernier-Colborne, C. Goutte, and S. Léger, "Improving cuneiform language identification with bert," in *Proceedings of the Sixth Workshop on NLP for Similar Languages, Varieties and Dialects*, 2019, pp. 17–25.

[15] N. M. Edan, "Cuneiform symbols recognition based on k-means and neural network," *AL-Rafidain Journal of Computer Sciences and Mathematics*, vol. 10, no. 1, pp. 195–202, 2013.

[16] C. Rest, D. Fisseler, F. Weichert, T. Somel, and G. G. Müller, "Illumination-based augmentation for cuneiform deep neural sign classification," *Journal on Computing and Cultural Heritage (JOCCH)*, vol. 15, no. 3, pp. 1–20, 2022.

[17] M. Li, T. Lv, J. Chen, L. Cui, Y. Lu, D. Florencio, C. Zhang, Z. Li, and F. Wei, "Trocr: Transformer-based optical character recognition with pre-trained models," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 37, no. 11, 2023, pp. 13 094–13 102.

[18] J. Baek, G. Kim, J. Lee, S. Park, D. Han, S. Yun, S. J. Oh, and H. Lee, "What is wrong with scene text recognition model comparisons? dataset and model analysis," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2019, pp. 4715–4723.

[19] L. Shi, Y. Zhang, J. Cheng, and H. Lu, "Skeleton-based action recognition with multi-stream adaptive graph convolutional networks," *IEEE Transactions on Image Processing*, vol. 29, pp. 9532–9545, 2020.

[20] K. Chen, L. Zhang, Z. Wang, S. Zhao, and Y. Zhou, "Skeleton-aware graph-based adversarial networks for human pose estimation from sparse imus," *ACM Transactions on Multimedia Computing, Communications and Applications*, vol. 21, no. 4, pp. 1–22, 2025.

[21] Y. Bengio, J. Louradour, R. Collobert, and J. Weston, "Curriculum learning," in *Proceedings of the 26th annual international conference on machine learning*, 2009, pp. 41–48.

[22] V. I. Spitkovsky, H. Alshawi, and D. Jurafsky, "From baby steps to leapfrog: How "less is more" in unsupervised dependency parsing," in *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, 2010, pp. 751–759.

[23] G. Hacohen and D. Weinshall, "On the power of curriculum learning in training deep networks," in *International conference on machine learning*. PMLR, 2019, pp. 2535–2544.

[24] X. Wang, Y. Chen, and W. Zhu, "A survey on curriculum learning," *IEEE transactions on pattern analysis and machine intelligence*, vol. 44, no. 9, pp. 4555–4576, 2021.

[25] CDLI contributors, "Home," https://cdli.earth/, jul 23 2025, [Online; accessed 2025-07-23]. [Online]. Available: https://cdli.earth/

[26] T. Chen, S. Kornblith, M. Norouzi, and G. Hinton, "A simple framework for contrastive learning of visual representations," in *International conference on machine learning*. PmLR, 2020, pp. 1597–1607.

[27] P. Khosla, P. Teterwak, C. Wang, A. Sarna, Y. Tian, P. Isola, A. Maschinot, C. Liu, and D. Krishnan, "Supervised contrastive learning," *Advances in neural information processing systems*, vol. 33, pp. 18 661–18 673, 2020.

[28] V. Yugay, K. Paliwal, Y. Cobanoglu, L. Sáenz, E. Gogokhia, S. Gordin, and E. Jiménez, "Stylistic classification of cuneiform signs using convolutional neural networks," *it-Information Technology*, vol. 66, no. 1, pp. 15–27, 2024.

[29] D. Kapon, M. Fire, and S. Gordin, "Shaping history: Advanced machine learning techniques for the analysis and dating of cuneiform tablets over three millennia," *arXiv preprint arXiv:2406.04039*, 2024.