

---

000 PROTONAP:  
001  
002 PROTOTYPE ALIGNMENT FOR CUNEIFORM SIGNS  
003  
004

005 **Anonymous authors**

006 Paper under double-blind review  
007  
008

009 ABSTRACT  
010

011 The cuneiform writing system served as the medium for transmitting knowledge  
012 in the ancient Near East for a period of over three thousand years. Cuneiform  
013 signs have a complex internal structure which is the subject of expert paleographic  
014 analysis, as variations in sign shapes bear witness to historical developments and  
015 transmission of writing and culture over time. However, prior automated techniques  
016 mostly treat sign types as categorical and do not explicitly model their highly varied  
017 internal configurations. In this work, we present an unsupervised approach for  
018 recovering the fine-grained internal configuration of cuneiform signs by leveraging  
019 powerful generative models and the appearance and structure of prototype font  
020 images as priors. Our approach, *ProtoSnap*, enforces structural consistency on  
021 matches found with deep image features to estimate the diverse configurations  
022 of cuneiform characters, snapping a skeleton-based template to photographed  
023 cuneiform signs. We provide a new benchmark of expert annotations and evaluate  
024 our method on this task. Our evaluation shows that our approach succeeds in  
025 aligning prototype skeletons to a wide variety of cuneiform signs. Moreover, we  
026 show that conditioning on structures produced by our method allows for generating  
027 synthetic data with correct structural configurations, significantly boosting the  
028 performance of cuneiform sign recognition beyond existing techniques, in particular  
029 over rare signs. We will release our code and data to the research community,  
030 foreseeing their use in a variety of applications in the digital humanities.

031 1 INTRODUCTION  
032

033 The earliest forms of decipherable scripts date back to the late 4<sup>th</sup> millennium BCE, with the invention  
034 of the cuneiform writing system in ancient Mesopotamia, which came to be used for a number  
035 of historically significant ancient languages such as Sumerian and Akkadian (Radner and Robson,  
036 2011; Streck, 2021). Cuneiform signs have complex internal structures which varied significantly  
037 across the eras, cultures, and geographic regions among which cuneiform writing was used. The  
038 study of these variations is part of a field called *paleography*, which is crucial for understanding the  
039 historical context of attested writing (Biggs, 1973; Homburg, 2021). However, while computational  
040 methods show promise for aiding experts in analyzing cuneiform texts (Bogacz and Mara, 2022),  
041 they are challenged by the vast variety of complex sign variants and their visual nature: Represented  
042 as wedge-shaped imprints in clay tablets which have often sustained physical damage, cuneiform  
043 appears as shadows on a non-uniform clay surface which may even be difficult for human experts to  
044 identify under non-optimal lighting conditions (Taylor, 2015).

045 Prior work has focused on digitization of cuneiform tablets at a coarse resolution, localizing and  
046 classifying signs from photographs of whole tablets (Dencker et al., 2020; Stötzner et al., 2023a).  
047 However, these methods treat sign types as categorical while neglecting sign-internal configurations  
048 of strokes in each character, which provides crucial information for identifying rare signs and  
049 distinguishing between sign variants. In this work, we aim to recover the fine-grained internal  
050 configuration of real cuneiform signs, given coarse-grained categorical information as input. In  
051 particular, our method is provided with a prototype image and its associated skeleton indicating the  
052 canonical structure of a sign, and aligns this structure to a target image depicting a corresponding real  
053 cuneiform sign. As illustrated in Figure 1, our technique is analogous to the laborious *hand copies*  
produced by expert Assyriologists; when applied to a tablet with existing character-level annotations,  
this outputs the outlines of signs in the style of the original document. Furthermore, we show that

054  
 055  
 056  
 057  
 058  
 059  
 060  
 061  
 062  
 063  
 064  
 065  
 066  
 067  
 068  
 069  
 070  
 071  
 072  
 073  
 074  
 075  
 076  
 077  
 078  
 079  
 080  
 081  
 082  
 083  
 084  
 085  
 086  
 087  
 088  
 089  
 090  
 091  
 092  
 093  
 094  
 095  
 096  
 097  
 098  
 099  
 100  
 101  
 102  
 103  
 104  
 105  
 106  
 107

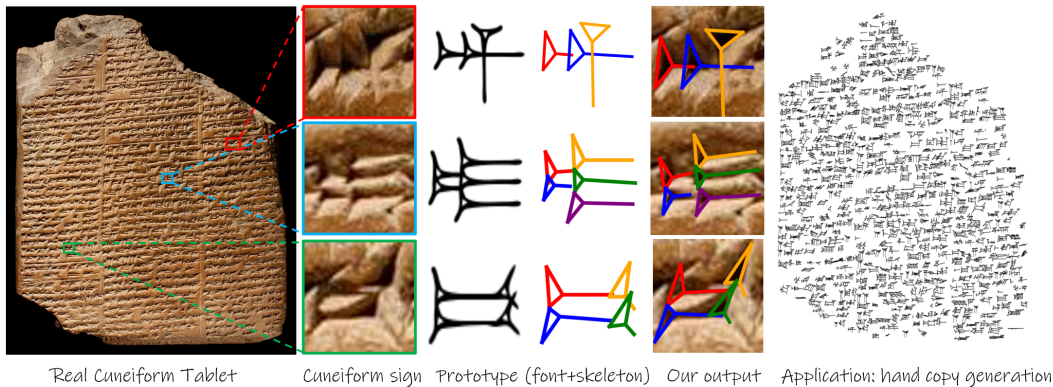


Figure 1: *ProtoSnap* applied to a full tablet by cropping each sign using existing bounding boxes (such as those depicted in unique colors), and matching prototypes of the signs (illustrated in the center). Our technique “snaps” the skeletons of the prototypes to the target images depicting real cuneiform signs. These aligned results can be used to produce an automatic digital *hand copy* (right). We also show that our approach can be used to boost performance of cuneiform sign recognition.

these aligned skeletons may be used to boost optical character recognition (OCR) performance, by training a generative model with structural conditioning as detailed below.

To this aim, we present *ProtoSnap*, an *unsupervised* approach leveraging deep diffusion features to snap a skeleton-based prototype to a target cuneiform sign, revealing its structure without requiring any labelled examples of real photographed signs. By using a fine-tuned generative model as a prior on the appearance of cuneiform images, and enforcing global and local consistency, we are able to localize the constituent strokes in real cuneiform images. We make use of the key insight that pairwise similarities between regions of the prototype and target images encode information about both coarse global alignment and fine-grained local deviations of each stroke from its canonical pose. Our technique distills this information with a multi-stage process performing global alignment followed by local refinement of stroke positions.

We provide a new benchmark of expert-annotated photographed signs for evaluation, and show that our system succeeds at identifying their internal structures, significantly outperforming generic correspondence matching techniques. We also show the downstream utility of *ProtoSnap* for automatic digitization of cuneiform texts, by using aligned prototype skeletons as a condition for a generative model to produce structurally-correct synthetic data to train cuneiform OCR. Our results show that this achieves state-of-the-art results on cuneiform sign recognition, particularly enhancing performance on rare signs where naive synthetic data generation struggles to produce instances of the correct sign.

Stated explicitly, our key contribution are:

- *ProtoSnap*, an unsupervised prototype alignment method capturing the structure of photographed cuneiform signs.
- A novel benchmark with expert annotations, and results showing that our method outperforms generic correspondence methods at this task.
- State-of-the-art OCR results for cuneiform when using synthetic data produced using our method’s alignments for conditional generation.

## 2 RELATED WORKS

**Machine learning (ML) for cuneiform.** Ancient texts provides a window into our history, but their decipherment and interpretation require painstaking work and expert knowledge of esoteric languages, complex writing systems, and historical context which serve as a barrier to their translation and analysis on scale. Due to the high societal value of these tasks and the scarcity of expert knowledge and time, machine learning promises to provide an invaluable aid for understanding the ancient world.



---

108 In the context of a number of works on ML applied to diverse ancient inscriptions (Assael et al., 2019;  
109 Yin et al., 2019; Huang et al., 2019; Luo et al., 2021; Hayon et al., 2024), the cuneiform script poses  
110 particular challenges. These include the nature of the physical writing media (indentations in textured  
111 and often damaged clay under various lighting conditions), and the diverse nature of cuneiform signs  
112 which changed over thousands of years of use in vast geographical regions.

113 Various approaches have been applied to modeling cuneiform signs for the purpose of downstream  
114 tasks such as optical character recognition. Some works apply recognition directly to image data (Bo-  
115 gacz et al., 2017; Dencker et al., 2020; Stötzner et al., 2023a), while others have treated cuneiform as  
116 structured graphs to recognize signs based on their internal configurations (Kriege et al., 2018; Chen  
117 et al., 2024). A handful of works have explicitly modeled cuneiform signs as compositions of wedges,  
118 though these mainly focus on segmentation from 3D meshes or localizing strokes on the bounding  
119 box level (Bogacz and Mara, 2022; Stötzner et al., 2023b; Hamplová et al., 2024). By contrast, our  
120 approach provides a *pixel-aligned* skeleton indicating the relative positions, sizes, and orientations of  
121 the strokes, and unlike prior works we operate exclusively on 2D photographs of cuneiform signs and  
122 without any strong supervision.

123 **Skeleton-based template alignment.** Our approach to inferring the configuration of cuneiform  
124 images as compositions of strokes by aligning a skeleton-based template bears partial similarity to  
125 various existing methods that typically operate over generic natural images.

126 Our method resembles template matching in that we use a template image (our font prototype) and  
127 search for relevant regions in the target image. While earlier approaches used naive comparisons of  
128 image intensities or low-level features when sliding the template across the target image (Ben-Arie  
129 and Rao, 1993; Cole et al., 2004; Kim et al., 2011), this is not robust to complex relations between  
130 the template and target. To handle these challenges, more recent works have adopted deep image  
131 features along with modeling complex non-rigid deformations (Oron et al., 2017; Talmi et al., 2017;  
132 Cheng et al., 2019; Gao and Spratling, 2022). Our method similarly searches for matches to our  
133 template using deep features and allowing for deformations, although we differ from conventional  
134 template matching in explicitly using the skeletonized graph structure of the template and matching  
135 each of its constituent strokes separately.

136 We also note similarity to pose estimation methods, as we infer the structure of a sign by localizing  
137 keypoints. Pose estimation methods align a graph of keypoints to an image, most commonly applied  
138 to a single category such as humans (Fang et al., 2022; Zheng et al., 2023), animals (Li and Lee, 2021;  
139 Yang et al., 2022), or vehicles (Reddy et al., 2018; López et al., 2019), where the same fixed graph  
140 applies to all instances. However, in our setting the number and connectivity of keypoints depends  
141 on the sign type under consideration. In this respect our method resembles category-agnostic pose  
142 estimation methods (Xu et al., 2022; Hirschorn and Avidan, 2023), though our input also includes a  
143 template image rather than only using an abstract graph.

144 In the context of images depicting text, a number of works address the problem of text spotting,  
145 which searches for matches to a given visual text representation in images (He et al., 2018; Huang  
146 et al., 2022; Ye et al., 2023). This may include alignment or dewarping of detected text, but does not  
147 typically explicitly leverage the internal shape of symbols as in our method. We also note several  
148 works performing transcript alignment using dense correspondence methods, which align visual text  
149 but do not explicitly handle character-internal structure (Hassner et al., 2013; 2016).

### 151 3 THE CUNEIFORM WRITING SYSTEM

152  
153 Cuneiform, one of the earliest known writing systems, was a logo-syllabic script indicating both units  
154 of sound and meaning with signs. Unlike the Latin alphabet which uses less than thirty basic letters  
155 to indicate sounds, cuneiform signs numbered upwards of one thousand unique types, which varied  
156 dramatically in their realizations across eras, languages, and geographical regions (Walker, 1987).  
157 See, for instance, the two right-most examples in Figure 5, both variants of the same sign AN from  
158 different eras. Scholars have attempted to collate lists of known signs and their variants (Labat and  
159 Malbran-Labat, 1988; Borger, 2003), and canonical representations of the most common variants for  
160 different time periods have been encoded in digital fonts.

161 Cuneiform was written on clay tablets by scribes using a stylus to create wedge-shaped impressions,  
also known as *strokes*, which combine to form signs. Scribes used styli with triangular edges to create

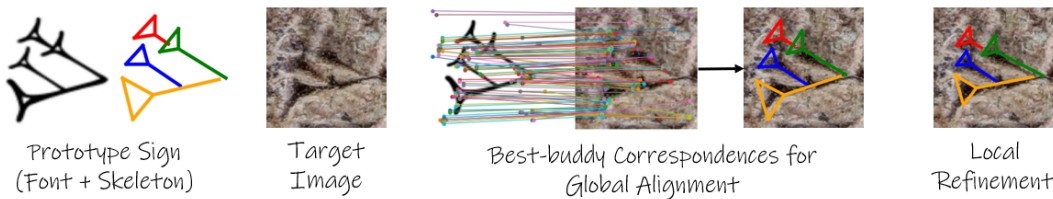


Figure 2: **Method Overview.** Given a prototype image with annotated skeleton and a target image of a real cuneiform sign, *ProtoSnap* first extracts *best-buddy* correspondences from deep diffusion features (extracted with our fine-tuned SD- $\Upsilon$  model), globally aligning the target image to the skeleton of the prototype. Our method then “snaps” the individual strokes into place with a local refinement stage by optimizing a per-stroke transform.

impressions on the moist clay in three possible directions: horizontal, vertical, or oblique (Cammarosano, 2014; Cammarosano et al., 2014). Various encoding schemes proposed to encode these strokes digitally (Bogacz and Mara, 2022); we adopt the four-keypoint scheme, treating all strokes as being composed of a triangular head indicated with three keypoints and a fourth keypoint indicating the stroke’s tail; see Figure 2 (second to left) for an example. We refer to the graph of these keypoints and the edges connecting them as a sign’s *skeleton*. Our method’s prototype input consists of both a rasterized font image along with its skeleton encoding the configuration of its strokes; we collect these skeletons via manual annotation as described in the appendix.

## 4 METHOD

Given a prototype consisting of a font image annotated with a skeleton composed of strokes and a target image of a real cuneiform sign, our system aligns the prototype skeleton to match the sign structure in the target image. An overview of this process is shown in Figure 2.

Our system consists of three steps: First, we calculate a semantically-adapted 4D similarity volume which encodes the pairwise feature similarities of each pair of regions in the two images. This similarity volume uses diffusion features to encapsulate the complex geometry and semantics of cuneiform images. We then calculate a global alignment between the prototype font image and the target image, using *best-buddies* sparse correspondences, extracted from the similarity volume, as a robust signal to fit this alignment. Finally, we perform local refinement to optimize the relative positions of each stroke and their internal configurations, when deviation from the canonical configuration is necessary. We describe each step in turn below, with further implementation details provided in the appendix.

### 4.1 SEMANTICALLY-ADAPTED 4D SIMILARITY VOLUME

Our system is based on the guiding assumption that local similarities between regions in the two images can be used to compute a structurally-consistent matching between the prototype sign structure and the target cuneiform sign. As a backbone for computing meaningful similarity scores, we use diffusion features (DIFT Tang et al. (2023)), which leverage the strong geometric and semantic understanding of a generative text-to-image model to represent image features for discriminative tasks. These features are calculated as intermediate activations the model’s denoising component, applied to the input image with added random noise. However, standard generative models are typically pretrained on natural images from the Internet, with cuneiform scans being out-of-distribution. We thus fine-tune the generative model Stable Diffusion (Rombach et al., 2022) on cuneiform image scans, which we indicate as SD- $\Upsilon$ , using the cuneiform sign name as its accompanying text prompt. We use this as our vision backbone for calculating DIFT features.

We apply DIFT with SD- $\Upsilon$  to our prototype and target images to obtain feature vector maps  $F^{(p)} = (\mathbf{f}_{i,j}^{(p)})_{i,j}$  and  $F^{(t)} = (\mathbf{f}_{i,j}^{(t)})_{i,j}$  respectively of unit-normalized feature vectors. Each feature map is a  $C \times H \times W$  tensor, where  $C$  is the number of feature dimensions and  $H$  and  $W$  are the spatial dimension of the feature map (lower-resolution than the original image, as each feature vector has a larger receptive field in the image). Using these features, we calculate the four-dimensional similarity

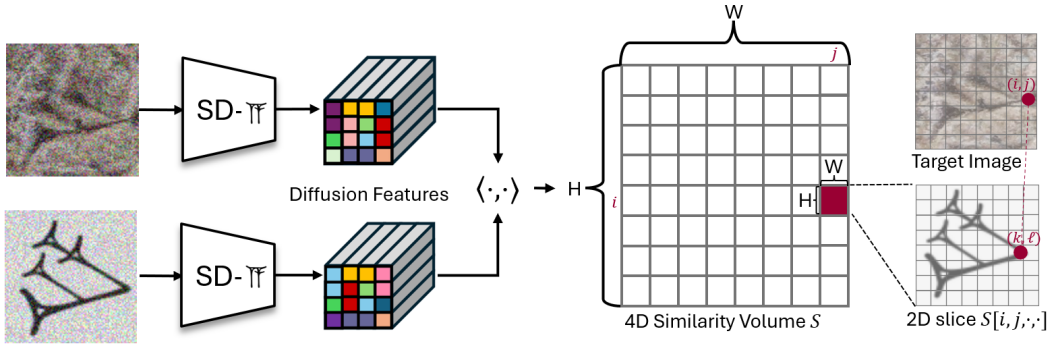


Figure 3: **DIFT-Based Best-Buddies Correspondences.** Noised images are passed through our fine-tuned denoising diffusion model  $\text{SD-}\Psi$  to extract deep Diffusion Features (DIFT), used to calculate the 4D similarity volume  $S$ . For each region  $(i, j)$  in the target image, we examine the 2D slice  $S[i, j, \cdot, \cdot]$ , and determine the indices  $(k, \ell)$  which maximize its value. Symmetrically, for each region  $(k, \ell)$  in the prototype we find the corresponding region in the target by maximizing the 2D slice  $S[\cdot, \cdot, k, \ell]$ . If these two regions correspond to each other, they are identified as *best buddies*.

volume  $S = (\mathbf{f}_{i,j}^{(p)} \cdot \mathbf{f}_{k,\ell}^{(t)})_{i,j,k,\ell}$ . This  $H \times W \times H \times W$  tensor, visualized in Figure 3, contains the pairwise cosine similarities between features encoding patches of the prototype and target images.

#### 4.2 GLOBAL ALIGNMENT FROM BEST-BUDDIES CORRESPONDENCES

While  $S$  provides a strong similarity measure, it does not encode geometric constraints on the overall matching between the two images. For instance, multiple regions in one image may all have high similarity scores with a single region in the other image, but mapping them all to the same target region will result in a degenerate solution. To robustly identify a sparse set of best-matching pairs of regions, we follow prior work (Oron et al., 2017; Drory et al., 2020) by identifying *best buddies*, defined as pairs of patches in the two images which are mutual nearest-neighbors according to their similarities scores in  $S$ . Formally, this is defined as pairs of coordinates  $(i, j)$  and  $(k, \ell)$  such that  $(i, j) = \arg \max_{i,j} S_{i,j,k,\ell}$  and  $(k, \ell) = \arg \max_{k,\ell} S_{i,j,k,\ell}$ . See Figure 3 for an illustration.

Using these sparse correspondences, we fit an affine transformation with least squares estimation, defining a global alignment  $G$  of the prototype to the target image. This transformation allows for basic deformations while preserving the overall structure of the prototype. We learn the parameters

$$G = \begin{bmatrix} g_{11} & g_{12} & g_{13} \\ g_{21} & g_{22} & g_{23} \\ 0 & 0 & 1 \end{bmatrix}$$

permitting scaling, rotation, and shear ( $g_{11}, g_{12}, g_{21}, g_{22}$ ) as well as translation ( $g_{13}, g_{23}$ ). This is applied in projective space  $\mathbb{P}^2$ , i.e. mapping a point  $\mathbf{v} = [x, y, 1]^T \in \mathbb{P}^2$  to the point  $\mathbf{v}' = G\mathbf{v} = [x', y', 1]^T$ .

To handle outliers in a robust manner, we perform the fitting with RANSAC. For further robustness, we incorporate a prior on inlier points being spread over the majority of the area of prototype and target images, by performing this procedure multiple times (including the stochastic calculation of DIFT features and correspondences from Section 4.1) and selecting the result with the best spread of inlier points across the relevant regions in the images, following Hassner et al. (2014) and as further described in the appendix.

#### 4.3 LOCAL REFINEMENT VIA SKELETON-BASED OPTIMIZATION

Our global alignment procedure can roughly align the prototype to the target image. However, as the target image was written by hand, each stroke’s location may deviate from the canonical relative position given by the prototype font image. Therefore, we introduce a local refinement stage, illustrated in Figure 4, which allows each stroke to move from the canonical prototype structure

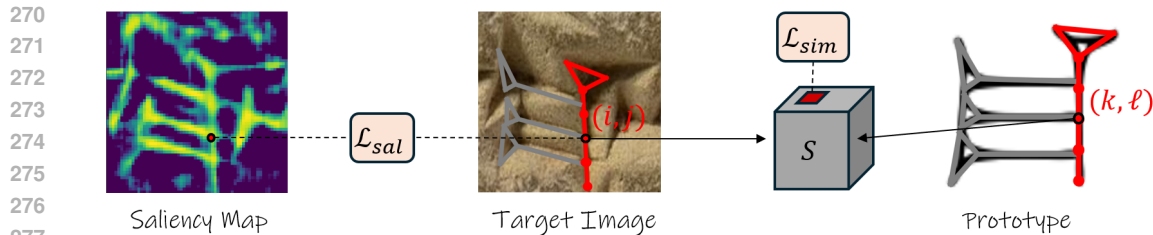


Figure 4: **Local Refinement via Skeleton-Based Optimization.** To adjust the positioning of individual strokes in a sign, our global alignment is followed by a local refinement stage which learns transformations for each stroke. The loss function encourages positioning on salient regions ( $\mathcal{L}_{sal}$ ) while semantically matching the corresponding regions in the prototype image, as measured by feature similarity ( $\mathcal{L}_{sim}$ ). For each stroke (exemplified by the stroke in red above), these objectives are calculated along points sampled from the skeleton (red dots above). The loss also includes a regularization term ( $\mathcal{L}_{reg}$ ) preventing excessive deviation from the global transformation.

and “snap” into place, while avoiding excessive deviations from the global structure representing the sign’s identity. We model each stroke’s deviation from the global alignment as a projective transformation, allowing for a higher degree of deformation than the affine global transformation. The local transformation of stroke  $i$  is parameterized by the matrix

$$P^{(i)} = I + \begin{bmatrix} p_{11}^{(i)} & p_{12}^{(i)} & p_{13}^{(i)} \\ p_{21}^{(i)} & p_{22}^{(i)} & p_{23}^{(i)} \\ p_{31}^{(i)} & p_{32}^{(i)} & 0 \end{bmatrix}$$

where  $I$  is the  $3 \times 3$  identity matrix. These are applied on top of the global transformation, resulting in per-stroke transformation of the form  $P^{(i)}G$ . As a projective transformation, this maps a point  $\mathbf{v} = [x, y, 1]^T \in \mathbb{P}^2$  to  $P^{(i)}G\mathbf{v} = [x', y', z']^T \sim [x'/z', y'/z', 1]^T$ . The  $p_{jk}^{(i)}$  are all initialized to zero (i.e. each local transformation is initialized as the identity).

We optimize the parameters  $p_{jk}^{(i)}$  via gradient descent with the loss function

$$\mathcal{L} = \lambda_{sim}\mathcal{L}_{sim} + \lambda_{sal}\mathcal{L}_{sal} + \lambda_{reg}\mathcal{L}_{reg}$$

where  $\lambda_{sim}, \lambda_{sal}, \lambda_{reg}$  are constant weights. We proceed to define each loss term.

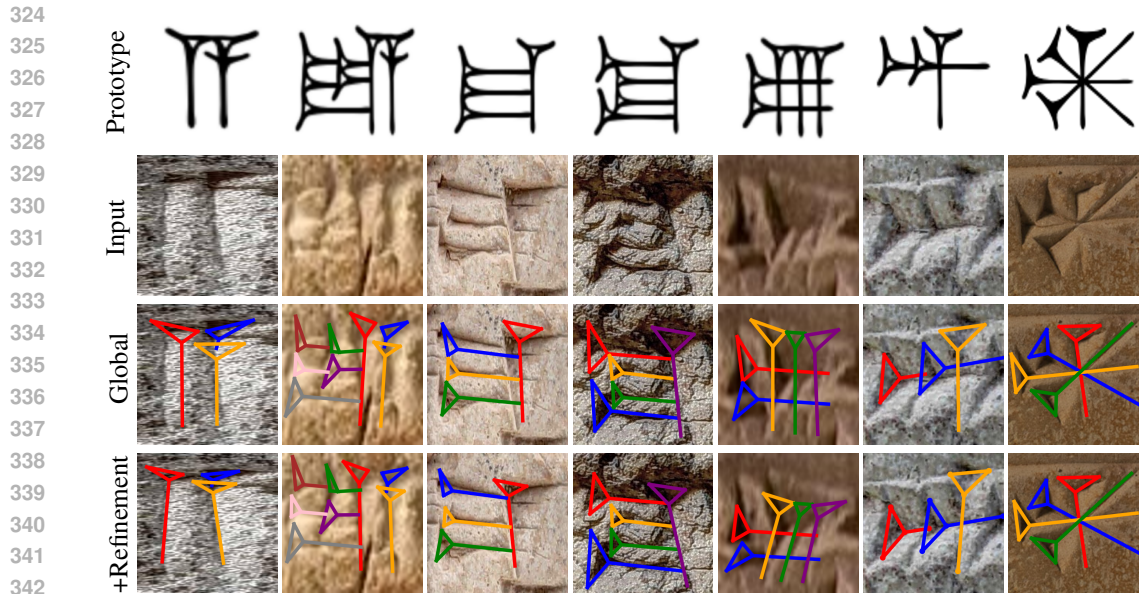
**Feature Similarity Loss  $\mathcal{L}_{sim}$ .** To encourage semantically-correct positioning of strokes, we define a loss to maximize feature similarities between matching points on the prototype and target images under the local transformation. Using the similarity volume  $S$  from 4.1, we sample points along the lines connecting skeleton keypoints in the prototype image, calculate their corresponding points under the current global and local transformations, and evaluate their similarity via  $S$  with a temperature-weighted softmax applied to each slice of  $S$  over the prototype image. This uses differentiable grid sampling to interpolate values of  $S$ , as  $S$  has a lower spatial resolution in comparison to the images.

**Saliency Map Loss  $\mathcal{L}_{sal}$ .** To encourage the strokes to cover *salient* regions (i.e. areas which appear to contain writing), we calculate a saliency map over the target image and use it to define a loss. The saliency map is calculated using the 4D similarity tensor  $S$  from Section 4.1; for each region in the target image, we calculate the difference between mean similarities to foreground (black) and background (white) regions in the prototype font image, and post-process by applying adaptive histogram equalization, scaling and setting low values to zero. This yields an approximate segmentation map of the cuneiform sign visible in the scanned image. The loss  $\mathcal{L}_{sal}$  is calculated as the mean value of the map sampled at points along the transformed skeleton, selecting points and using temperature-scaled differentiable grid sampling as in Section 4.1.

**Regularization Loss  $\mathcal{L}_{reg}$ .** To avoid invalid solutions that over-optimize the previous objectives, we add a regularization term that penalizes excessive deformations and solutions that stray from the boundaries of the image. This is defined as  $\mathcal{L}_{reg} = \mathcal{L}_{L1} + \mathcal{L}_{ob}$ . L1 regularization loss is given by

$$\mathcal{L}_{L1} = \frac{1}{N} \sum_{i,j,k} \left| p_{jk}^{(i)} \right| = \frac{1}{N} \sum_i \|P^{(i)} - I\|_{L1}$$





343 Figure 5: **Qualitative alignment results**, aligning the prototypes (first row) to target cuneiform  
344 images (second row). We demonstrate the results after performing global alignment (third row), and  
345 the final result after local refinement (fourth row). As illustrated above, the global alignment stage  
346 provides a coarse placement of the prototype template, while the refinement stage allows each stroke  
347 to slightly diverge from the original prototype, resulting in more accurate alignments.

348  
349 where  $N$  is the number of strokes and  $I$  is the  $3 \times 3$  identity matrix. This penalizes local transforma-  
350 tions which greatly deviate from the identity.

351 The out-of-bounds loss  $\mathcal{L}_{oob}$  is defined as zero if all transformed keypoints are within the image  
352 boundaries, otherwise as the maximum absolute difference between each transformed coordinate  
353 and the image bounds. This models the soft constraint that all keypoints must lie within the image,  
354 handling edge cases where the global transformation pushes part of a stroke outside the image bounds.  
355

## 356 5 EXPERIMENTS

357  
358 We conduct various experiments, evaluating our method both directly on our benchmark of expert-  
359 annotated real photographed signs (Section 5.1) and by testing it on a downstream OCR benchmark  
360 (Section 5.2). We also present qualitative results (Figure 5 and appendix), which further highlight the  
361 complexity and unique challenges of the task and setting addressed in our work. Finally, we discuss  
362 limitations of our approach (Section 5.3). Further implementation details and results are provided in  
363 the appendix.  
364

### 365 5.1 ALIGNMENT EVALUATION

366  
367 To evaluate the quality of alignment, we curate a test set of ground-truth (GT) alignments from  
368 manual annotations by expert archeologists. Provided with photographs of cuneiform signs from  
369 the eBL dataset Cobanoglu et al. (2024), the experts annotated the position of strokes by indicating  
370 groups of four keypoints. Images were marked for exclusion from the test set if they were poor  
371 quality or show sign variants differing from the corresponding prototype font images. In total, our  
372 test set contains 272 images with expert annotations, covering 25 different sign types. A breakdown  
373 of this test set and more details on our annotation procedure are provided in the appendix.  
374

375 To quantify performance on this benchmark, we compare the predicted and GT positions of stroke  
376 keypoints; given a fixed distance threshold, we consider a predicted keypoint as a true positive  
377 prediction if it is within the given threshold of a GT keypoint. We report F1 scores for several distance  
thresholds (along with precision and recall in the appendix).



Method	F1@20	F1@30	F1@40
SIFT (Lowe, 1999) + RANSAC	2.56%	3.78%	5.01%
DINOv2 (Oquab et al., 2024)	12.33%	21.88%	31.04%
DINOv2 + RANSAC	16.12%	28.19%	37.93%
DIFT (Tang et al., 2023)	16.14%	25.80%	33.79%
DIFT + RANSAC	13.13%	21.96%	30.15%
Ours (w/o refinement)	20.49%	35.70%	48.80%
Ours (full)	<b>26.86%</b>	<b>41.71%</b>	<b>52.17%</b>

Table 1: **Alignment evaluation**, measuring keypoint localization at various distance thresholds. We compare against several generic correspondence matching baselines, including a geometry-based method (SIFT) and deep feature-based methods (DINOv2, DIFT). As illustrated above, our method significantly outperforms these baselines. Furthermore, our local refinement stage provides a performance boost beyond learning simply a global transform.

In Table 1 we compare our solution to generic correspondence matching techniques. We compare against a traditional geometry-based baseline that computes SIFT (Lowe, 1999) features and aligns the images using RANSAC. We also compare against two deep feature-based techniques (DINOv2 (Oquab et al., 2024), DIFT Tang et al. (2023)), applied by assigning each keypoint to the corresponding point of maximum similarity in the target image. In the case of DIFT, diffusion features are calculated with our fine-tuned SD- $\mathbb{F}$ .

We see that *ProtoSnap* significantly outperforms the baseline methods at localization, as our solution explicitly considers the complex structure of the sign. We also report performance of our approach without local refinement (*i.e.* performing the first step of global alignment alone); we see that the final step of local optimization indeed achieves better alignment as reflected in our metrics. This may also be seen visually in Figure 5, illustrating the overall global alignment and more precise results yielded by local refinement. In the appendix, we show an ablation study of the different components of our system (such as using fine-tuned SD- $\mathbb{F}$  and each of our loss terms). These results provide further motivation for the design of our full system. We also provide additional qualitative results.

## 5.2 LEARNING OCR FROM ALIGNED DATA

We show the downstream benefit of our approach by using *ProtoSnap* alignments to produce structurally-diverse synthetic training data for an OCR system. As cuneiform signs are highly diverse, with number and arrangement of strokes varying significantly between eras, regions, and individuals, a model trained to produce synthetic data conditioned on sign type alone may struggle to depict the correct configurations of signs, particularly rare signs with few to no attestations in the existing data. Furthermore, conditioning on sign type does not allow to specify the exact variant of the sign, resulting in generation of the most prevalent variant in the data. As we show, conditioning on a skeleton-based structures provides control over the generated fine-grained structures yielding synthetic data which better captures the real configurations. To demonstrate the benefit of our approach for the downstream task of OCR, we align prototype skeletons to a set of cuneiform images from the eBL dataset (Cobanoglu et al., 2024), and fine-tune ControlNet (Zhang et al., 2023) to generate new cuneiform images using such skeletons as a condition, instead of a text prompt. This model, denoted as CN- $\mathbb{F}$ , allows us to produce new cuneiform sign images with any input structure, not limited to a list of predefined categories or specific structural configurations. We then use this model to generate synthetic training data, as shown in Figure 6. Further details about the model training are provided in the appendix.

We examine the benefit of this generated data for learning cuneiform sign classification. Dencker et al. (2020) report sign classification performance on the CSDD dataset when training a Resnet18 (He et al., 2016) model with supervision from the CSDD training set alone. We compare this to augmenting the training set with the CN- $\mathbb{F}$ -generated synthetic data described above. As an additional baseline, we also compare with using SD- $\mathbb{F}$  (see Section 4.1) generations for augmenting sign types which uses categorical sign names as a textual condition rather than structural conditions. We report classification accuracy following Dencker et al. (2020); as this dataset is highly imbalanced, we also

432  
433  
434  
435  
436  
437  
438  
439  
440  
441  
442  
443  
444  
445  
446  
447  
448  
449  
450  
451  
452  
453  
454  
455  
456  
457  
458  
459  
460  
461  
462  
463  
464  
465  
466  
467  
468  
469  
470  
471  
472  
473  
474  
475  
476  
477  
478  
479  
480  
481  
482  
483  
484  
485

Training Data	Accuracy		Balanced Accuracy	
	All	Rare	All	Rare
CSDD* (Dencker et al., 2020)	58.43%	25.84%	34.57%	16.45%
+SD- $\Upsilon$ data	61.56%	38.56%	39.02%	31.13%
+CN- $\Upsilon$ data	<b>64.14%</b>	<b>53.17%</b>	<b>43.57%</b>	<b>39.98%</b>

Table 2: Sign classification performance when training on previously collected real data (CSDD), and with added data generated using our fine-tuned diffusion model (SD- $\Upsilon$ ) or ControlNet trained using alignments from *ProtoSnap* (CN- $\Upsilon$ ). Our solution demonstrates improved OCR results, with structural augmentations showing increased performance relative to direct unconditional image generation, especially on rare signs (signs that have less than 50 occurrences in the real train set). \* denotes our reproduced model, as further described in the appendix.

report balanced accuracy and performance on rare sign types (signs with less than 50 occurrences in the real training set). Results are summarized in Table 2. As seen there, augmenting the CSDD dataset with synthetic data significantly improves classification performance, with structurally-conditioned data from CN- $\Upsilon$  providing a significant boost over synthetic data from SD- $\Upsilon$ . These results reflect that structurally-controlled generation with CN- $\Upsilon$  guarantees generation of the correct sign variant, while SD- $\Upsilon$  struggles to produce correct configurations from the sign category alone, as seen in Figure 6.

### 5.3 LIMITATIONS

We note various limitations of our work (visualized in Figure 7). Our alignment procedure requires a canonical sign image and will still fail if the scan displays a structurally different variant of the sign in question. Additionally, our method may fail under extreme deformations, or on low-quality tablets or scans which cannot be feasibly interpreted. Future work might investigate how to calculate a confidence measure to detect such failure cases.

## 6 CONCLUSION

We have proposed the *ProtoSnap* method for estimating the internal structure of cuneiform signs without any direct supervision, by harnessing pairwise comparisons of deep image features in regions of photographed cuneiform images and skeleton-based prototypes. We have curated expert annotations to provide a new benchmark for this task, and have shown that our method significantly outperforms generic correspondence-based techniques. Beyond its direct application for paleographic analysis, we have also shown our method’s utility for the downstream task of OCR, by using aligned skeletons for conditional synthetic data generation to achieve SOTA performance on cuneiform sign recognition.

We foresee a range of applications and possible extensions of our work. The automatic curation and quantification of precise sign configurations and variants shows potential for performing paleographic analysis on scale, with implications for dating tablets, identifying genres and authors, and understanding the development of cuneiform writing across time and space. Our method could be used to automate the production of hand copies illustrating the contents of a tablet in their original context and handwriting style, currently produced manually by experts. While we have focused on the case of single sign scans, we foresee future work extending these results to lines of text, as our method could be incorporated into a pipeline including text detection and localization of individual signs applied either to images or directly to 3D scans of inscriptions. Additionally, our method represents a step towards the goal of compiling a dataset of *allographs* (sign variants) for each sign, indicating how individual scribes executed them within a given period, city, archive, scribal school, or even personal handwriting styles. Finally, we have shown the downstream utility of our method for producing structurally-diverse training data for OCR, showing promise for the digitization of cuneiform inscriptions. We hope that our work will spur future research on using the internal structure of glyphs in complex scripts such as cuneiform to advance downstream tasks.

486  
487  
488  
489  
490  
491  
492  
493  
494  
495  
496  
497  
498  
499  
500  
501  
502  
503  
504  
505  
506  
507  
508  
509  
510  
511  
512  
513  
514  
515  
516  
517  
518  
519  
520  
521  
522  
523  
524  
525  
526  
527  
528  
529  
530  
531  
532  
533  
534  
535  
536  
537  
538  
539

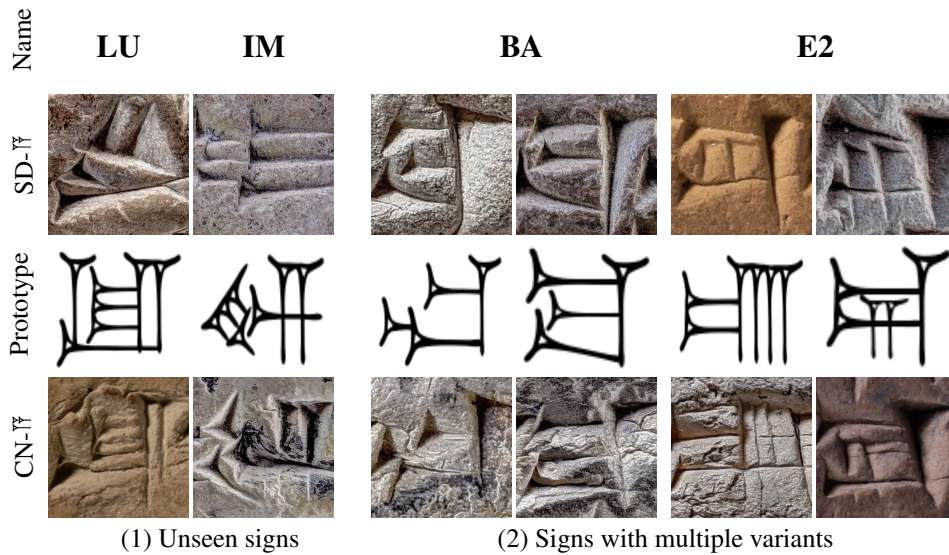


Figure 6: We demonstrate the benefit of producing structurally-controlled synthetic data (denoted as CN- $\Upsilon$  above), in comparison to text-conditional generation (denoted as SD- $\Upsilon$ ), using two different scenarios: (1) Text-conditional generation cannot succeed in generating signs unseen during training, while our method can correctly adhere to the structural conditioning provided as input, even for rare or unseen signs. (2) A text-conditional model often generates the sign variation most prevalent during training (*e.g.*, the variants on the right sides above), while our method can generate different variants, yielding a more diverse synthetic set for training downstream models.

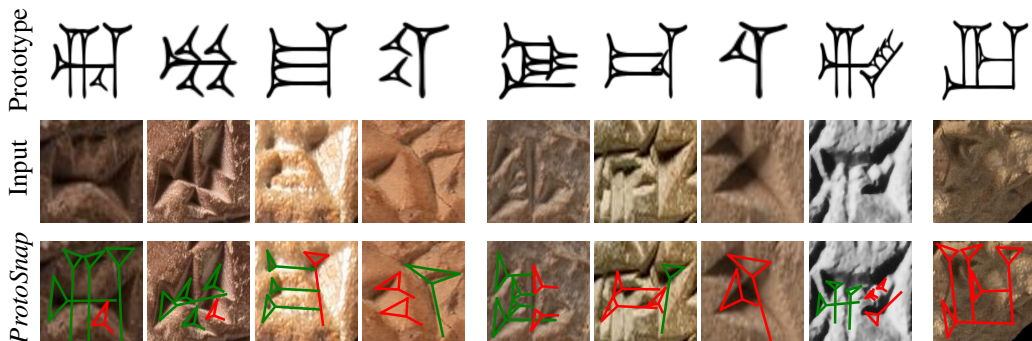


Figure 7: **Limitations of our method**, illustrating examples with significant deformations from the prototype skeleton (left) and structurally different sign variants (middle) and corrupt sign image (right). We visualize correctly-aligned strokes in green, and misaligned strokes in red.

540  
541  
542  
543  
544  
545  
546  
547  
548  
549  
550  
551  
552  
553  
554  
555  
556  
557  
558  
559  
560  
561  
562  
563  
564  
565  
566  
567  
568  
569  
570  
571  
572  
573  
574  
575  
576  
577  
578  
579  
580  
581  
582  
583  
584  
585  
586  
587  
588  
589  
590  
591  
592  
593

## REFERENCES

- Karen Radner and Eleanor Robson, editors. *The oxford handbook of cuneiform culture*. Oxford University Press, Oxford, 2011. doi: 10.1093/oxfordhb/9780199557301.001.0001. URL <https://doi.org/10.1093/oxfordhb/9780199557301.001.0001>.
- Michael P. Streck, editor. *Sprachen des Alten Orients*. wbg Academic, Darmstadt, 4., überarbeitete und aktualisierte auflage edition, 2021.
- Robert D Biggs. On regional cuneiform handwritings in third millennium mesopotamia. *Orientalia*, 42:39–46, 1973.
- Timo Homburg. Paleocodage—enhancing machine-readable cuneiform descriptions using a machine-readable paleographic encoding. *Digital Scholarship in the Humanities*, 36(Supplement\_2):ii127–ii154, 2021.
- Bartosz Bogacz and Hubert Mara. Digital assyriology—advances in visual cuneiform analysis. *Journal on Computing and Cultural Heritage (JOCCH)*, 15(2):1–22, 2022.
- Jon Taylor. *Wedge Order in Cuneiform: a Preliminary Survey*, page 1–30. PeWe-Verlag, Gladbeck, 2015.
- Tobias Dencker, Pablo Klinkisch, Stefan M Maul, and Björn Ommer. Deep learning of cuneiform sign detection with weak supervision using transliteration alignment. *Plos one*, 15(12):e0243039, 2020.
- Ernst Stötzner, Timo Homburg, and Hubert 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*, pages 1680–1688, 2023a.
- Yannis Assael, Thea Sommerschild, and Jonathan Prag. Restoring ancient text using deep learning: a case study on Greek epigraphy. In *Empirical Methods in Natural Language Processing*, pages 6369–6376, 2019.
- Xusen Yin, Nada Aldarrab, Beáta Megyesi, and Kevin Knight. Decipherment of historical manuscript images. In *2019 International Conference on Document Analysis and Recognition (ICDAR)*, pages 78–85. IEEE, 2019.
- Shuangping Huang, Haobin Wang, Yongge Liu, Xiaosong Shi, and Lianwen Jin. Obc306: A large-scale oracle bone character recognition dataset. In *2019 International Conference on Document Analysis and Recognition (ICDAR)*, pages 681–688. IEEE, 2019.
- Jiaming Luo, Frederik Hartmann, Enrico Santus, Regina Barzilay, and Yuan Cao. Deciphering under-segmented ancient scripts using phonetic prior. *Transactions of the Association for Computational Linguistics*, 9:69–81, 2021.
- Offry Hayon, Stefan Münger, Ilan Shimshoni, and Ayellet Tal. Arcaid: Analysis of archaeological artifacts using drawings. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 7264–7274, 2024.
- Bartosz Bogacz, Maximilian Klingmann, and Hubert Mara. Automating transliteration of cuneiform from parallel lines with sparse data. In *2017 14th IAPR International Conference on Document Analysis and Recognition (ICDAR)*, volume 1, pages 615–620. IEEE, 2017.
- Nils M Kriege, Matthias Fey, Denis Fisseler, Petra Mutzel, and Frank Weichert. Recognizing cuneiform signs using graph based methods. In *International Workshop on Cost-Sensitive Learning*, pages 31–44. PMLR, 2018.
- Yizhou Chen, Anxiang Zeng, Qingtao Yu, Kerui Zhang, Cao Yuanpeng, Kangle Wu, Guangda Huzhang, Han Yu, and Zhiming Zhou. Recurrent temporal revision graph networks. *Advances in Neural Information Processing Systems*, 36, 2024.
- Ernst Stötzner, Timo Homburg, Jan Philipp Bullenkamp, and Hubert Mara. R-cnn based polygonal wedge detection learned from annotated 3d renderings and mapped photographs of open data cuneiform tablets. 2023b.

- 
- 594 Adéla Hamplová, Avital Romach, Josef Pavlíček, Arnošt Veselý, Martin Čejka, David Franc, and  
595 Shai Gordin. Cuneiform stroke recognition and vectorization in 2d images. *Digital Humanities*  
596 *Quarterly*, 18(1), 2024. URL [https://www.digitalhumanities.org/dhq/vol18/](https://www.digitalhumanities.org/dhq/vol18/1/000733/000733.html)  
597 [1/000733/000733.html](https://www.digitalhumanities.org/dhq/vol18/1/000733/000733.html).
- 598  
599 Jezekiel Ben-Arie and K Raghunath Rao. A novel approach for template matching by nonorthogonal  
600 image expansion. *IEEE Transactions on Circuits and Systems for Video Technology*, 3(1):71–84,  
601 1993.
- 602  
603 Luke Cole, David Austin, Lance Cole, et al. Visual object recognition using template matching. In  
604 *Australian conference on robotics and automation*, 2004.
- 605  
606 Hae Yong Kim et al. Ciratefi: An rst-invariant template matching with extension to color images.  
607 *Integrated Computer-Aided Engineering*, 18(1):75–90, 2011.
- 608  
609 Shaul Oron, Tali Dekel, Tianfan Xue, William T Freeman, and Shai Avidan. Best-buddies simi-  
610 larity—robust template matching using mutual nearest neighbors. *IEEE transactions on pattern*  
611 *analysis and machine intelligence*, 40(8):1799–1813, 2017.
- 612  
613 Itamar Talmi, Roey Mechrez, and Lihi Zelnik-Manor. Template matching with deformable diversity  
614 similarity. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*,  
615 pages 175–183, 2017.
- 616  
617 Jiaxin Cheng, Yue Wu, Wael AbdAlmageed, and Premkumar Natarajan. Qatm: Quality-aware  
618 template matching for deep learning. In *Proceedings of the IEEE/CVF Conference on Computer*  
619 *Vision and Pattern Recognition*, pages 11553–11562, 2019.
- 620  
621 Bo Gao and Michael W Spratling. Robust template matching via hierarchical convolutional features  
622 from a shape biased cnn. In *The International Conference on Image, Vision and Intelligent Systems*  
623 *(ICIVIS 2021)*, pages 333–344. Springer, 2022.
- 624  
625 Hao-Shu Fang, Jiefeng Li, Hongyang Tang, Chao Xu, Haoyi Zhu, Yuliang Xiu, Yong-Lu Li, and  
626 Cewu Lu. Alphapose: Whole-body regional multi-person pose estimation and tracking in real-time.  
627 *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2022.
- 628  
629 Ce Zheng, Wenhan Wu, Chen Chen, Taojiannan Yang, Sijie Zhu, Ju Shen, Nasser Kehtarnavaz, and  
630 Mubarak Shah. Deep learning-based human pose estimation: A survey. *ACM Computing Surveys*,  
631 56(1):1–37, 2023.
- 632  
633 Chen Li and Gim Hee Lee. From synthetic to real: Unsupervised domain adaptation for animal  
634 pose estimation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern*  
635 *recognition*, pages 1482–1491, 2021.
- 636  
637 Yuxiang Yang, Junjie Yang, Yufei Xu, Jing Zhang, Long Lan, and Dacheng Tao. Apt-36k: A  
638 large-scale benchmark for animal pose estimation and tracking. *Advances in Neural Information*  
639 *Processing Systems*, 35:17301–17313, 2022.
- 640  
641 N Dinesh Reddy, Minh Vo, and Srinivasa G Narasimhan. Carfusion: Combining point tracking and  
642 part detection for dynamic 3d reconstruction of vehicles. In *Proceedings of the IEEE conference*  
643 *on computer vision and pattern recognition*, pages 1906–1915, 2018.
- 644  
645 Javier García López, Antonio Agudo, and Francesc Moreno-Noguer. Vehicle pose estimation via  
646 regression of semantic points of interest. In *2019 11th International Symposium on Image and*  
647 *Signal Processing and Analysis (ISPA)*, pages 209–214. IEEE, 2019.
- 648  
649 Lumin Xu, Sheng Jin, Wang Zeng, Wentao Liu, Chen Qian, Wanli Ouyang, Ping Luo, and Xiaogang  
650 Wang. Pose for everything: Towards category-agnostic pose estimation. In *European conference*  
651 *on computer vision*, pages 398–416. Springer, 2022.
- 652  
653 Or Hirschorn and Shai Avidan. Pose anything: A graph-based approach for category-agnostic pose  
654 estimation. *arXiv preprint arXiv:2311.17891*, 2023.



- 
- 648 Tong He, Zhi Tian, Weilin Huang, Chunhua Shen, Yu Qiao, and Changming Sun. An end-to-end  
649 textspotter with explicit alignment and attention. In *Proceedings of the IEEE conference on*  
650 *computer vision and pattern recognition*, pages 5020–5029, 2018.
- 651
- 652 Mingxin Huang, Yuliang Liu, Zhenghao Peng, Chongyu Liu, Dahua Lin, Shenggao Zhu, Nicholas  
653 Yuan, Kai Ding, and Lianwen Jin. Swintextspotter: Scene text spotting via better synergy between  
654 text detection and text recognition. In *proceedings of the IEEE/CVF conference on computer vision*  
655 *and pattern recognition*, pages 4593–4603, 2022.
- 656 Maoyuan Ye, Jing Zhang, Shanshan Zhao, Juhua Liu, Tongliang Liu, Bo Du, and Dacheng Tao.  
657 Deepsolo: Let transformer decoder with explicit points solo for text spotting. In *Proceedings of the*  
658 *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 19348–19357, 2023.
- 659
- 660 Tal Hassner, Lior Wolf, and Nachum Dershowitz. Ocr-free transcript alignment. In *2013 12th*  
661 *International Conference on Document Analysis and Recognition*, pages 1310–1314. IEEE, 2013.
- 662 Tal Hassner, Lior Wolf, Nachum Dershowitz, Gil Sadeh, and Daniel Stökl Ben-Ezra. Dense corre-  
663 spondences and ancient texts. *Dense Image Correspondences for Computer Vision*, pages 279–295,  
664 2016.
- 665
- 666 Christopher Bromhead Fleming Walker. *Cuneiform*, volume 3. Univ of California Press, 1987.
- 667
- 668 R. Labat and F. Malbran-Labat. *Manuel d'Épigraphie Akkadienne*. Paris, sixth edition edition, 1988.
- 669
- 670 R. Borger. *Mesopotamisches Zeichenlexicon*. Alter Orient Und Altes Testament: Veröffentlichungen  
671 Zur Kultur Und Geschichte Des Alten Orients Und Des Alten Testaments. Ugarit Verlag, Münster,  
672 2003.
- 673 Michele Cammarosano. The cuneiform stylus. *Mesopotamia: Rivista di Archeologia, Epigrafia e*  
674 *Storia Orientale Antica*, 69:53–90, 2014.
- 675 Michele Cammarosano, Gerfrid G. W. Müller, Denis Fisseler, and Frank Weichert. Schriftmetrologie  
676 des keils: Dreidimensionale analyse von keileindrücken und handschriften. *Die Welt des Orients*,  
677 44(1):2–36, 2014.
- 678
- 679 Luming Tang, Menglin Jia, Qianqian Wang, Cheng Perng Phoo, and Bharath Hariharan. Emergent  
680 correspondence from image diffusion. *Advances in Neural Information Processing Systems*, 36:  
681 1363–1389, 2023.
- 682 Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-  
683 resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF confer-*  
684 *ence on computer vision and pattern recognition*, pages 10684–10695, 2022.
- 685
- 686 Amnon Drory, Tal Shomer, Shai Avidan, and Raja Giryes. Best buddies registration for point clouds.  
687 In *Proceedings of the Asian Conference on Computer Vision*, 2020.
- 688
- 689 Tal Hassner, Liav Assif, and Lior Wolf. When standard ransac is not enough: cross-media visual  
690 matching with hypothesis relevancy. *Machine Vision and Applications*, 25:971–983, 2014.
- 691 Yunus Cobanoglu, Luis Sáenz, Ilya Khait, and Enrique Jiménez. Sign detection for cuneiform  
692 tablets. *it - Information Technology*, 66(1):28–38, 2024. doi:doi:10.1515/itit-2024-0028. URL  
693 <https://doi.org/10.1515/itit-2024-0028>.
- 694
- 695 David G Lowe. Object recognition from local scale-invariant features. In *Proceedings of the seventh*  
696 *IEEE international conference on computer vision*, volume 2, pages 1150–1157. Ieee, 1999.
- 697
- 698 Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov,  
699 Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, Mahmoud Assran, Nicolas  
700 Ballas, Wojciech Galuba, Russell Howes, Po-Yao Huang, Shang-Wen Li, Ishan Misra, Michael  
701 Rabbat, Vasu Sharma, Gabriel Synnaeve, Hu Xu, Hervé Jegou, Julien Mairal, Patrick Labatut,  
Armand Joulin, and Piotr Bojanowski. Dinov2: Learning robust visual features without supervision,  
2024.

---

702 Lvmin Zhang, Anyi Rao, and Maneesh Agrawala. Adding conditional control to text-to-image  
703 diffusion models, 2023.

704  
705 Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image  
706 recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*,  
707 pages 770–778, 2016.

708 Eugen Rusakov, Turna Somel, Gernot A. Fink, and Gerfrid G.W. Müller. Towards query-by-  
709 expression retrieval of cuneiform signs. In *2020 17th International Conference on Frontiers in*  
710 *Handwriting Recognition (ICFHR)*, pages 43–48, 2020. doi: 10.1109/ICFHR2020.2020.00019.

## 711 712 APPENDIX

### 713 714 A EXPERIMENTAL DETAILS

715  
716 Below we provide additional experimental details. Our code is also provided (zipped in the supple-  
717 mentary material). For all experiments described below, a single A5000 GPU was used. Running the  
718 method on a single image takes about 1 minute.

#### 719 720 A.1 IMAGE AND FONT INFORMATION

721 For all of our tests, we use RGB images with resolution  $512 \times 512$ , resizing images as needed.

722  
723 Font images are rendered from the *Santakku* and *SantakkuM* fonts, designed by Sylvie Vanséverin  
724 and available on the Hethitologie Portal Mainz. For a uniform appearance, the white margins of the  
725 image are cropped, than 10 pixels of white margins are added to each side, and finally the image is  
726 resized to  $512 \times 512$  resolution.

#### 727 728 A.2 MODEL DETAILS

729  
730 We use the `CompVis/stable-diffusion-v1-4` checkpoint as our base Stable Diffusion  
731 model. We fine-tune this on eBL classification train dataset (Cobanoglu et al., 2024), for 50K  
732 iterations with batch size of 4, learning rate of  $10^{-5}$  and Adam optimizer. For textual prompts,  
733 we use a unique code for each sign type indicated in the eBL dataset. This fine-tuned Stable  
734 Diffusion model, `SD- $\mathbb{F}$` , is used for our DIFT feature calculations as well as in the synthetic  
735 data generation for our OCR application. For tests using ControlNet, we fine-tune the base  
736 `illyasviel/sd-controlnet-openpose` checkpoint, on 932 samples with their paired align-  
737 ment created by *ProtoSnap*. We trained it for 20K iteration, with batch size of 4, learning rate of  $10^{-5}$   
738 and Adam optimizer. This fine-tuned ControlNet, `CN- $\mathbb{F}$` , uses another fine-tuned Stable Diffusion  
739 model, trained with the same parameters as `SD- $\mathbb{F}$` , but without a prior from a textual prompt, using  
740 the same prompt for all signs: "cuneiform single ancient icon".

741 For DIFT feature calculations, we average an ensemble of results on four random noises, sampled  
742 at timestamp  $t = 261$ , following Tang et al. (2023). We concatenate features from the second and  
743 third upscaling U-Net layers, bilinearly interpolated to the same spatial resolution, yielding a set of  
744  $64 \times 64$  feature vectors of dimension 1920.

#### 745 746 A.3 GLOBAL ALIGNMENT DETAILS

747  
748 To fit our global alignment, we apply RANSAC with 2000 iterations. At each iteration, 5 correspon-  
749 dences are used to fit a least-squares affine transformation, with a distance threshold of 50 pixels used  
750 to identify outliers. The transformation with the greatest number of inliers is returned.

751 As a high-quality set of correspondences should explain the relevant regions in both of the images, we  
752 incorporate a prior on inlier points being spread across the images, following Hassner et al. (2014). In  
753 particular, we perform the above procedure 8 times, and assign each result a score using the convex  
754 hulls of the inlier points in the prototype and scanned cuneiform sign images. For the prototype  
755 image, we calculate the proportion  $p_{proto}$  of the prototype font foreground contained within the  
convex hull of inlier points. For the scanned cuneiform sign image, we calculate the proportion  $p_{scan}$

---

756 of the total area of the image covered by the convex hull of inlier points. Finally, we select the result  
757 with maximum score  $p_{proto} \cdot p_{scan}$ , to encourage the global transformation to be based on matches  
758 between regions covering most of the scan and prototype font.  
759

#### 760 A.4 LOCAL REFINEMENT DETAILS

761 Our total loss uses coefficients  $\lambda_{sim} = 1.0$ ,  $\lambda_{sal} = 3 \times 10^{-4}$ , and  $\lambda_{reg} = 10^{-4}$ .

762 To calculate the saliency map used for the saliency loss, we compute differences in mean similarities  
763 as described in the main paper, apply CLAHE histogram equalization with clip limit 10.0 and tile  
764 grid size (2, 2), set values below the mean to zero, and scale values to the range [0, 1]. This yields a  
765 scalar field of resolution  $64 \times 64$ .  
766

767 For both semantic similarity and saliency losses, at each iteration values are sampled at both each  
768 of the keypoints in the skeleton, and at 8 randomly-sampled points along each line segment in the  
769 skeleton connecting keypoints. Loss values are averaged over all of these points. The sampled points  
770 are sampled uniformly from the lines in the prototype images, and then transformed using the current  
771 global and local transformations to obtain corresponding points in the target scanned cuneiform sign  
772 image. Loss values are computed with differentiable grid sampling, using bilinear interpolation over  
773 scalar fields (the similarity values in respective slices of  $S$ , and saliency map values) each of which is  
774 passed through a softmax with fixed temperature parameter 100.  
775

776 To perform optimization, we apply gradient descent for 100 iterations with learning rate 0.01 and  
777 Adam optimizer, updating the the parameters of the local transformations of all strokes.  
778

#### 779 A.5 DATASET DETAILS

780 Both the training and the test datasets are taken from the eBL classification dataset Cobanoglu et al.  
781 (2024) which consists of scanned imaged of cuneiform signs. The dataset was originally split to train  
782 and test by source tablet. The train set included 47,820 sign images, representing 387 sign types and  
783 was used to train SD- $\overline{\text{f}}$ . The test set was used to select the 272 images for our test set (as described in  
784 5.1), focusing on signs for which we have an available prototype, and the samples in the set match the  
785 prototype structure variant. The final test set consists of 272 samples from 25 different signs, varying  
786 from 2 strokes per sign to 8.  
787

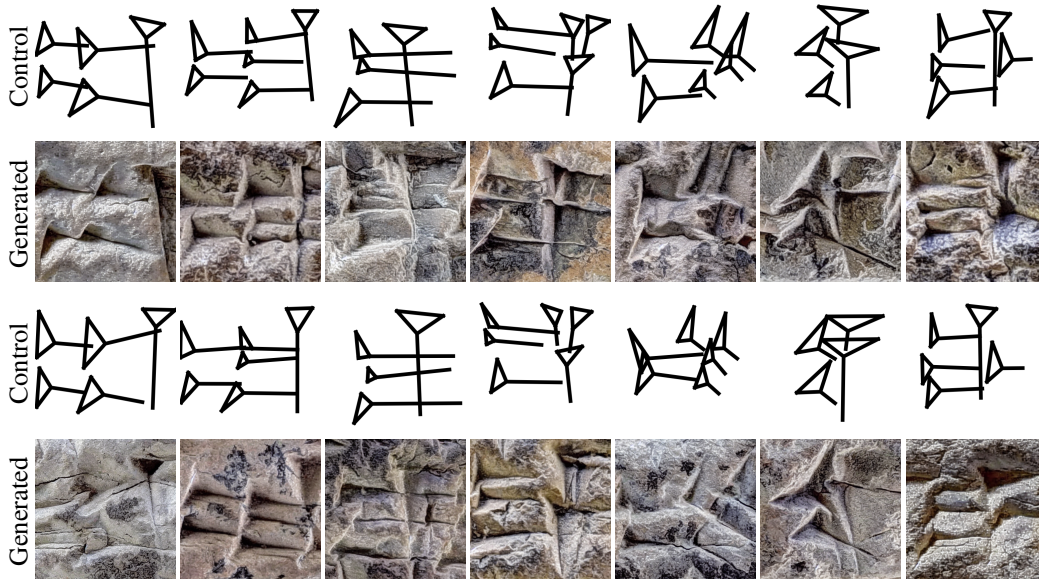
788 The full dataset comprises around 40% from the Neo-Babylonian period (1000–600 BC), around  
789 20% from the Neo-Assyrian (1000-609 BCE), and less than 10% from the following periods: Ur  
790 III (2100–2002 BCE), Old Babylonian (2002-1595 BCE), Old Assyrian (1950–1850 BCE), Mid-  
791 dle Babylonian (1500–1000 BCE), Late Babylonian (600 BC–100 AD), Persian (539-331 BCE),  
792 Hellenistic (331-141 BCE), Parthian (141 BCE-100 CE). The dataset represents the Akkadian and  
793 Sumerian languages used at those eras.  
794

#### 795 A.6 OCR EXPERIMENT DETAILS

796 For the OCR experiment we have generated 50 samples per each sign in the test set (180 signs in  
797 total), using our fine-tuned SD- $\overline{\text{f}}$  described above. In addition, we used CN- $\overline{\text{f}}$  to generate 50 samples  
798 for each sign we have available prototype (124 signs in total). To better mimic human handwriting,  
799 we augmented the skeletons by applying small random transformation on the entire skeleton and on  
800 each stroke individually, creating a diverse set of controls for each sign. Figure 8 shows examples of  
801 such generated data. Both generations were done using 50 inference steps and classifier-free guidance  
802 scale 7.5.  
803

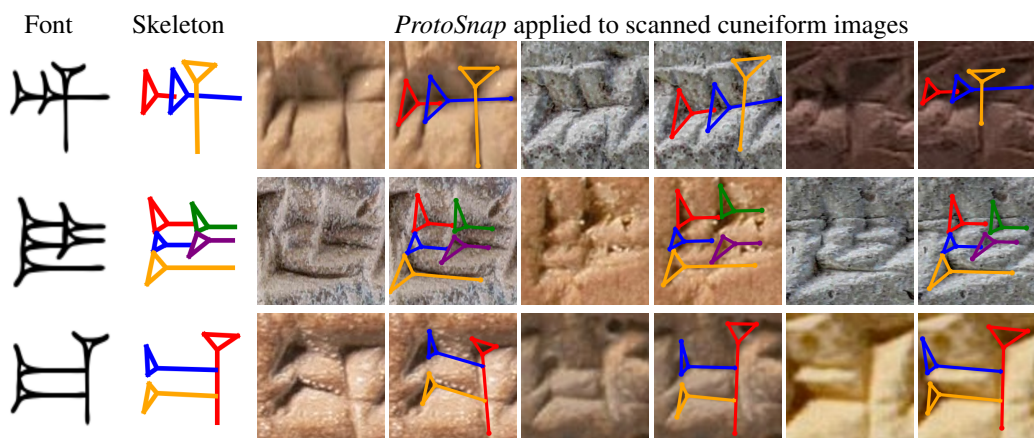
804 For the experiment we used 6205 samples from the CSDD data (consisting of data from all URLs  
805 which were not broken in the dataset), 3299 samples in the train set and 2906 in the test. Training  
806 ResNet18 on this dataset alone was done for 10 epochs, using batch size of 64, learning rate of  $10^{-3}$   
807 and Adam optimizer. Training with generated data (both from SD- $\overline{\text{f}}$  and CN- $\overline{\text{f}}$ ) was done for 8  
808 epochs, and then fine-tuned 10 more on real data only, using batch size of 64, learning rate of  $5^{-4}$   
809 and Adam.

810  
811  
812  
813  
814  
815  
816  
817  
818  
819  
820  
821  
822  
823  
824  
825  
826  
827  
828  
829



830 Figure 8: Examples for data generated using our fine-tuned ControlNet model CN-ṽ, where the control is an image of a prototype sign (with two different added transformations).  
831

832  
833  
834  
835  
836  
837  
838  
839  
840  
841  
842  
843  
844  
845  
846



847 Figure 9: Examples of *ProtoSnap* applied on photographed cuneiform signs of varying structure, illumination conditions and degrees of intactness.  
848  
849

850  
851  
852  
853  
854

Note that our reproduced baseline on CSDD achieves slightly higher accuracy than reported by Dencker et al. (2020), but the test dataset includes URLs which are no longer operational which may contribute to this slight discrepancy in results.

## 855 B ADDITIONAL RESULTS

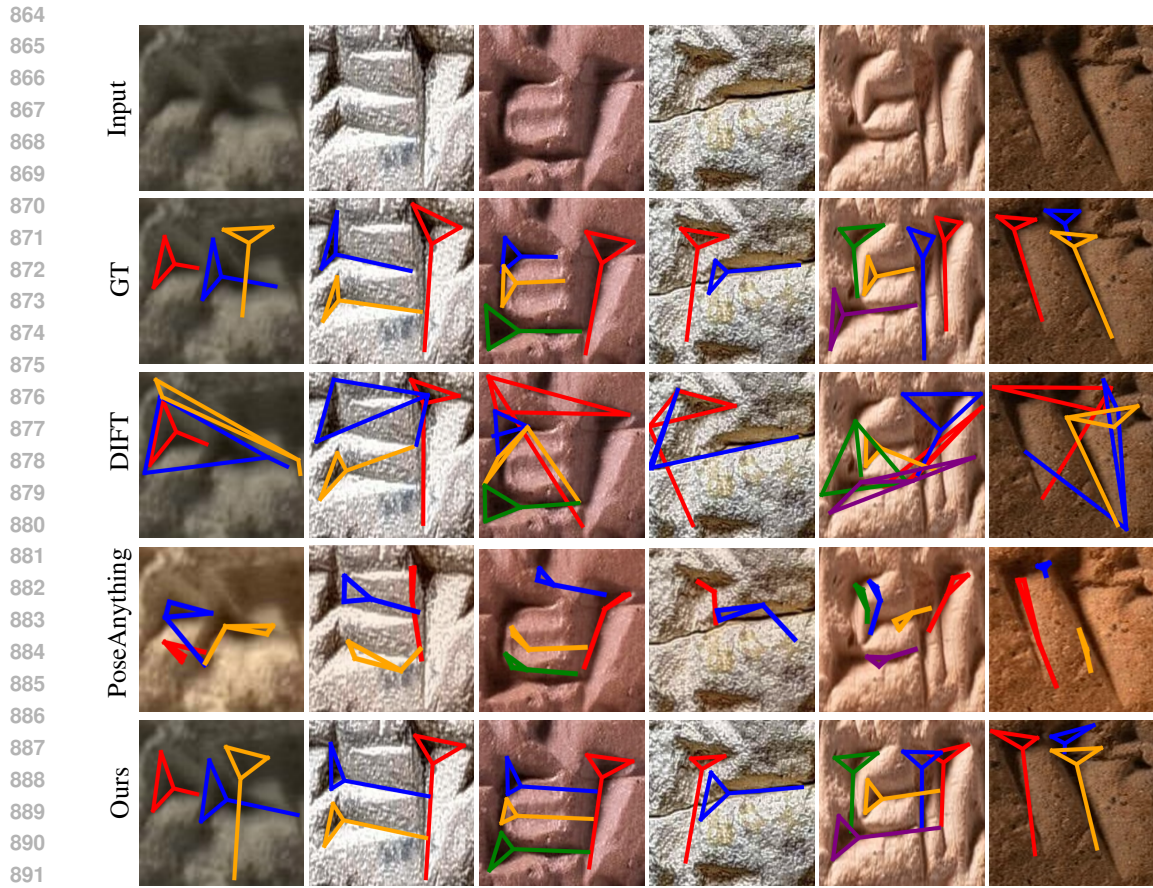
856  
857  
858  
859  
860

Figure 9 shows additional examples, illustrating *ProtoSnap* applied to various samples of the same sign. Figure 10 shows *ProtoSnap* results on our manually annotated test set, compared to the baseline of applying DIFT directly (assigning each keypoint to the region of maximal feature similarity).

861  
862  
863

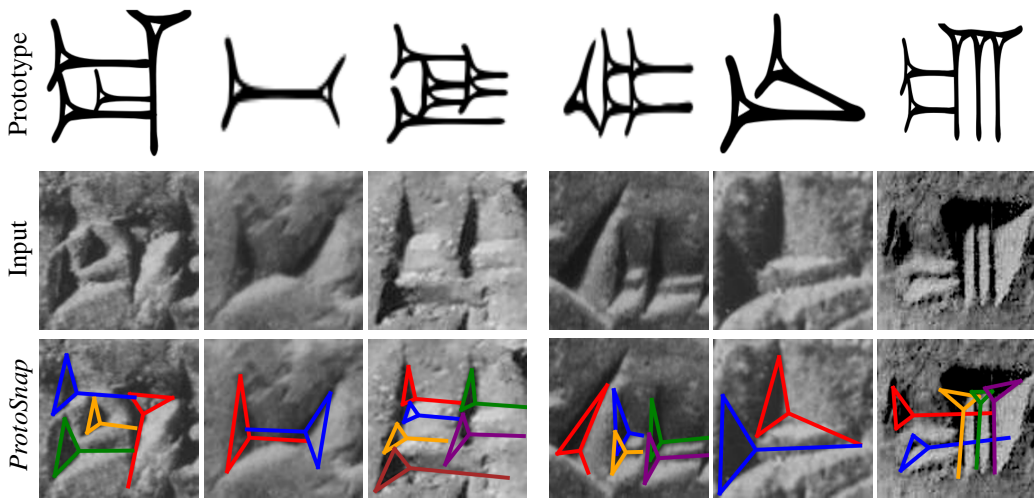
Figure 11 shows *ProtoSnap* results on a new, previously unseen dataset, JOCCH (Rusakov et al., 2020), which contains signs from the Hittite language, as opposed to the Akkadian and Sumerian from which the training and test set are composed. Those results show that our method is robust and can be generalizable to other usages of the cuneiform writing system.





892  
893  
894  
895  
896  
897  
898  
899  
900  
901  
902

Figure 10: Results of *ProtoSnap* on our manually annotated test set, with DIFT and PoseAnything (Hirschorn and Avidan, 2023) shown for comparison. We can see that our method produces alignments which are much closer to expert annotations and is generally less sensitive to outliers.



914  
915  
916  
917

Figure 11: *ProtoSnap* applied on images from a different dataset and language (Hittite), showing that the method is robust and generalizable to various usages of cuneiform writing system. The 3 images on the left show signs from types (names) unseen in the training data, further emphasizing the generalizability of the method.



Method	<i>threshold = 20</i>			<i>threshold = 40</i>		
	Precision	Recall	F1	Precision	Recall	F1
SIFT (Lowe, 1999) + RANSAC	2.59%	2.53%	2.56%	5.49%	4.61%	4.91%
DINOv2 (Oquab et al., 2024)	12.42%	12.25%	12.33%	32.09%	30.05%	31.04%
DINOv2 + RANSAC	16.22%	16.01%	16.12%	38.47%	37.42%	37.93%
DIFT (Tang et al., 2023)	16.18%	16.10%	16.14%	34.32%	33.27%	33.79%
DIFT + RANSAC	13.15%	13.11%	13.13%	30.43%	29.88%	30.15%
Ours (w/o refinement)	20.58%	20.39%	20.49%	49.16%	48.44%	48.80%
Ours (full)	<b>26.89%</b>	<b>26.83%</b>	<b>26.86%</b>	<b>52.40%</b>	<b>51.94%</b>	<b>52.17%</b>

Table 3: Precision and recall metrics for the alignment evaluation, on top of F1 metric presented in the main paper, at two distance thresholds.

Sign Name	# Samples	# Strokes	F1@20	F1@30	F1@40
ME	20	2	20.00%	35.62%	46.25%
A	19	3	26.53%	37.49%	48.46%
IGI	18	3	13.43%	25.93%	36.34%
AN	19	3	29.39%	49.34%	62.72%
UD	18	3	16.20%	34.26%	46.53%
GISH	19	3	25.44%	39.91%	52.63%
MA	16	4	30.47%	44.34%	55.27%
EN	5	5	18.00%	34.00%	45.49%
IR	17	5	23.38%	38.38%	45.88%
IB	15	5	33.83%	52.16%	62.66%
HA	3	5	21.67%	46.67%	55.00%
UR	16	5	30.00%	44.69%	55.46%
RI	13	5	26.54%	37.31%	51.15%
RU	15	5	24.00%	38.49%	47.83%
DIM2	2	5	15.00%	41.21%	56.22%
DI	6	5	37.08%	47.08%	54.17%
U2	9	5	26.39%	39.99%	50.56%
DIB	3	6	19.44%	33.33%	44.44%
SA	3	6	40.23%	47.18%	57.56%
GI	1	7	39.29%	50.00%	51.72%
DA	1	7	28.57%	39.29%	48.15%
KA	12	7	27.08%	45.24%	57.14%
ZI	10	7	26.42%	40.71%	48.38%
A2	4	8	31.64%	44.92%	54.69%
ZE2	8	8	40.04%	53.32%	61.33%

Table 4: **Performance breakdown by sign type.** We report alignment performance of our model over the different annotated signs. We also provide the number of strokes in each sign (#Strokes) and number of samples of the sign in the test set (#Samples).

## B.1 ADDITIONAL METRICS

Table 3 shows precision and recall metrics for the alignment evaluation, on top of F1 metric presented in the main paper. Table 4 show alignment evaluation breakdown per signs, and also provides the number of samples in the test set per sign.

## B.2 ABLATION STUDY

We demonstrate the effect of key parts of our system by ablating them and evaluating performance on our test set. In particular, we ablate the following:

- Use of our fine-tuned SD- $\Upsilon$  (rather than base Stable Diffusion)
- Use of *best-buddies* correspondences for computing the global transformation (rather than using all correspondences between prototype image regions and the best-matching regions in the target image according to DIFT)
- Convex hull scoring to select a global alignment from multiple attempts
- Each of the three loss terms in our full loss function
- Including sampled points along the edges of the prototype skeleton for calculating losses with grid sampling (rather than only using the skeleton keypoints)

As seen in Table 5, each of these ablations has a negative effect on performance, showing the benefit of all components of our system used together.

	F1@20	F1@30	F1@40
<i>ProtoSnap</i> (ours)	<b>26.86%</b>	<b>41.71%</b>	<b>52.17%</b>
–SD- $\Upsilon$	19.01%	31.89%	41.25%
–best buddies	25.00%	39.55%	49.70%
–convex hull scoring	26.52%	39.79%	48.03%
– $\mathcal{L}_{sim}$	26.67%	40.77%	49.60%
– $\mathcal{L}_{sal}$	20.85%	37.03%	49.76%
– $\mathcal{L}_{reg}$	26.10%	37.88%	46.06%
–edge sampling	25.97%	40.16%	50.92%

Table 5: Ablation study results, demonstrating differences in performance when removing key parts of our system. Our full method shows the best performance as reflected by the test set metrics above.

## C ANNOTATION DETAILS

Our expert annotations were performed by archaeologists who participated in this research.

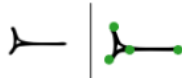
Below, we provide further details on our annotations collected via crowdsourcing, used to annotate keypoints in prototype font images and in scanned cuneiform signs. We then connected the keypoints manually ourselves, creating the prototype skeleton.

### C.1 IRB APPROVAL, PARTICIPANT SOURCING, AND COMPENSATION

Our annotation tasks, approved by our institution’s IRB, were conducted on the Amazon Mechanical Turk (MTurk) crowdsourcing platform. We published our tasks for MTurk workers with at least 1000 completed HITs (MTurk tasks) and a HIT approval rate of at least 95%. Workers were compensated \$0.25 for each font annotation, corresponding to the duration of this task.

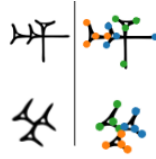
### C.2 ANNOTATION TASK INSTRUCTIONS

*In this task, you will indicate keypoints on ancient character ("cuneiform") to indicate the location of each stroke. Please indicate each stroke with four keypoints as shown here:*



*If there are multiple strokes, please indicate each stroke in a separate color using four keypoints per stroke, as in these examples:*

1026  
1027  
1028  
1029  
1030  
1031  
1032  
1033  
1034  
1035  
1036  
1037  
1038  
1039  
1040  
1041  
1042  
1043  
1044  
1045  
1046  
1047  
1048  
1049  
1050  
1051  
1052  
1053  
1054  
1055  
1056  
1057  
1058  
1059  
1060  
1061  
1062  
1063  
1064  
1065  
1066  
1067  
1068  
1069  
1070  
1071  
1072  
1073  
1074  
1075  
1076  
1077  
1078  
1079



*Make sure the four keypoints are in the locations as shown above – three indicating the corners of the stroke’s triangular head, and one indicating the end of its tail.*

*Use the point tool to place points on the requested target(s) of interest: Four each stroke in the character; place four keypoints of the same color; using your mouse to click on each keypoint. Use the four keypoints described in the instructions: for each stroke, three points indicating the corners of the stroke’s triangular head, and one indicating the end of its tail. Make sure to indicate every stroke seen in the glyph.*