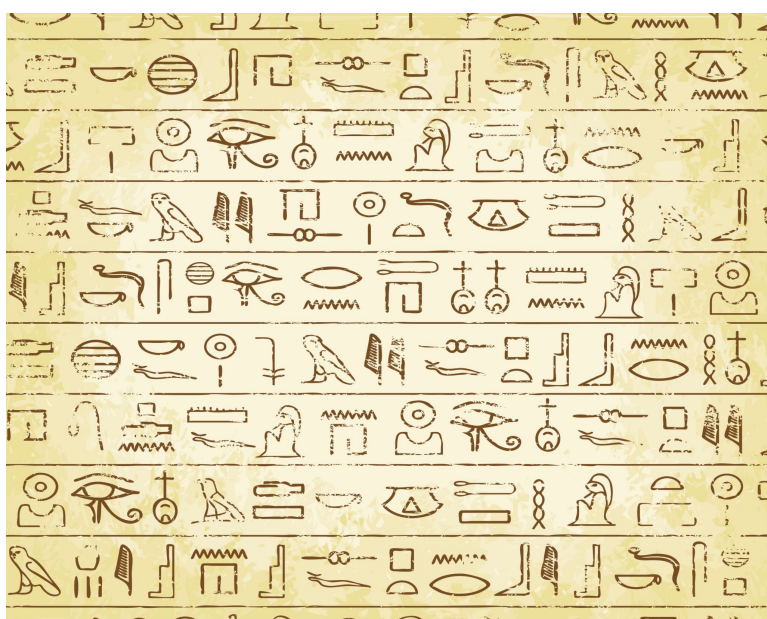


Raport

System do Analizy Fonetycznych Hieroglifów Egipskich



Autorzy:
Bartosz Słowik, Paweł Samuła

Wstęp

Prezentowany raport opisuje system stworzony do odczytywania i konwersji fonetycznych hieroglifów egipskich na litery alfabetu łacińskiego. System ten, opracowany z myślą o ułatwieniu interpretacji starożytnych symboli, wykorzystuje narzędzia informatyczne w swojej funkcjonalności.

Opis Funkcji Systemu

System został zaprojektowany do analizy fonetycznych hieroglifów egipskich, które stanowią jeden z najstarszych zapisanych języków. Celem systemu jest przetłumaczenie tych znaków na litery alfabetu łacińskiego.

Wyzwania Techniczne

Różnorodność Symboli: Hieroglify egipskie cechuje duża różnorodność formy i znaczenia, co wymaga analizy wielu kształtów i interpretacji.

Złożoność Struktury: Hieroglify składają się z pojedynczych znaków i kombinacji symboli tworzących złożone wyrażenia, co wymaga zrozumienia ich kontekstu i relacji między nimi.

Brak Standaryzacji: Hieroglify ewoluowały przez długi czas, co sprawia, że ich interpretacja wymaga uwzględnienia historycznych zmian w formie i znaczeniu.

Ograniczenia Techniczne: Przetwarzanie dużych zbiorów danych graficznych hieroglifów stanowi wyzwanie ze względu na ich graficzną złożoność.

Podziały Hieroglifów

Hieroglify egipskie dzielą się na kilka kategorii, takich jak piktogramy, ideogramy, fonetyczne i determinanty. Każda z tych kategorii ma swoją unikalną funkcję i znaczenie w systemie pisma hieroglificznego.

Korzystanie z programu

Interfejs graficzny (GUI) programu umożliwia łatwe ładowanie obrazów hieroglifów i oferuje sekcję "Przykłady" z przetłumaczonymi hieroglifami.

Działanie programu

Wczytywanie i Przetwarzanie Obrazu:

Na początku działania programu funkcja **load_and_process_image** zajmuje się przygotowaniem obrazu do analizy. Ten etap obejmuje wczytanie obrazu z pliku, skalowanie go do wymaganego rozmiaru, konwersję na odcienie szarości oraz odwrócenie kolorów. Przetworzenie obrazu w ten sposób jest fundamentalne, ponieważ umożliwia lepsze dopasowanie wzorców hieroglifów w kolejnych etapach.

```
def load_and_process_image(file_path, upscale_factor=2.0):  
    # Read the image  
    image = io.imread(file_path)  
    # Upscale the image  
    image = rescale(image, scale=upscale_factor, anti_aliasing=True, channel_axis=2)  
    # Convert the image to grayscale and invert  
    image = rgb2gray(image)  
    image = 1 - image  
  
    return image
```

Wykrywanie Hieroglifów:

Po przetworzeniu obrazu, zadziała kolejna metoda: **detect_glyphs**. Jej zadaniem jest wykrywanie hieroglifów na obrazie. Wykorzystuje ona algorytm dopasowania wzorców do porównania przetworzonego obrazu z wcześniej zdefiniowanymi wzorcami hieroglifów. Dzięki identyfikacji i lokalizacji hieroglifów na obrazie, funkcja ta jest w stanie zwrócić ich dokładne położenie. Program używa progu tolerancji, aby rozróżnić między prawidłowym dopasowaniem a przypadkowym szumem na obrazie.

```
def detect_glyphs(template, image, tolerance, glyph_name):
    if image is None:
        return

    result = match_template(image, template)
    coordinates = []

    ij = np.unravel_index(np.argmax(result), result.shape)
    x, y = ij[::-1]

    while result[y, x] > tolerance:
        coordinates.append((x, y, glyph_name, result[y, x]))

        # Define the size of the area to zero out based on a slightly smaller
        # fraction of the detected letter size
        size = max(template.shape) // 4 # Adjust as needed for zeroing out
        zero_out_size = int(size * 0.8) # Reduce the zero-out area by 20%

        # Zero out a region around the maximum
        result[max(0, y - zero_out_size):min(result.shape[0], y + zero_out_size +
1),
                max(0, x - zero_out_size):min(result.shape[1], x + zero_out_size +
1)] = 0

        ij = np.unravel_index(np.argmax(result), result.shape)
        x, y = ij[::-1]

    return coordinates
```

Procesowanie i Wizualizacja Wyników:

Ostatnim etapem jest funkcja **process_image**, która integruje wyniki wykrywania hieroglifów i przekształca je w interpretowalny tekst. Ta funkcja koordynuje działania wcześniejszych etapów, sortując wykryte hieroglify i łącząc je w czytelny ciąg znaków. Po zakończeniu tego procesu, program wykorzystuje `plot_detected_glyphs` do stworzenia wizualizacji, gdzie wykryte hieroglify są zaznaczone na oryginalnym obrazie. Dodatkowo, wyświetlany jest tekst reprezentujący zinterpretowane hieroglify.

```
def process_image(image, templates):
    for key, (template, threshold) in templates.items():
        if template.shape[0] > image.shape[0] or template.shape[1] > image.shape[1]:
            scaled_size = (min(template.shape[0], image.shape[0] - 1),
                           min(template.shape[1], image.shape[1] - 1))
            templates[key] = (resize(template, scaled_size), threshold)

    found_glyphs = []
    for glyph, (template, threshold) in templates.items():
        found_glyphs.extend(detect_glyphs(template, image, threshold, glyph))

    ROW_HEIGHT = 166 # The height of each row in pixels

    found_glyphs.sort(key=lambda glyph: (glyph[1] // ROW_HEIGHT, glyph[0]))
    detected_letters = [glyph[2] for glyph in found_glyphs]
    word = ''.join(detected_letters)
    print("Interpreted Word:", word)

    plot_detected_glyphs(image, found_glyphs, templates, word)
```

Interfejs

System wykorzystuje bibliotekę `scikit-image` do analizy i przetwarzania plików `jpg` z hieroglifami, obejmując:

Definicję Wzorca Hieroglifu: Użytkownik definiuje wzorzec hieroglifu do rozpoznania.

Porównanie Wzorca z Obrazem: System analizuje obraz w poszukiwaniu dopasowań.

Lokalizacja Dopasowania: Algorytmy lokalizują najlepsze dopasowanie wzorca na obrazie.

Proces Analizy Hieroglifów

Identyfikacja Znaków: System rozpoznaje poszczególne hieroglify, analizując ich kształt i rozmiar.

Analiza Kontekstu: Zrozumienie kontekstu występowania hieroglifów i ich wzajemnych relacji.

Konwersja na Grafemy: Przekształcenie hieroglifów na odpowiadające im grafemy alfabetu łacińskiego.

Input: Obraz przedstawiający hieroglify.

Output: Konwertowane znaki na litery alfabetu łacińskiego.

Bibliografia

Hieroglyphic typewriter_-

<https://discoveringegypt.com/hieroglyphic-typewriter/>

List of egyptian hieroglyphs

https://en.wikipedia.org/wiki/List_of_Egyptian_hieroglyphs

Hieroglyphic dataset -

<https://www.kaggle.com/datasets/waleedumer/egyptian-hieroglyphic-s-datasets>