International Journal of Modern Research in Engineering and Technology (IJMRET)
www.ijmret.org Volume 8 Issue 10 ǁ December 2023.

Matching of Archaeologically Fragmented 2d Objects Using Image Processing and Artificial Intelligence Methods

Gurel Yildiz[1], Nevcihan Duru[2]
[1](Department of Computer Engineering / Kocaeli University, Turkey)
[2](Department of Computer Engineering / Kocaeli Health and Technology University, Turkey)

ABSTRACT: *Restoring 2D fragmented historical ruins is a time-consuming and difficult process for archaeologists. During the reassembly process, archaeologists manually compare the remains to decide on correct matches. In this article, we present a new method to accelerate and automate the process of correctly matching and combining two 2D parts. In general, in previous studies, it has been observed that image edge segmentation is approached by assuming that the image edge segments obtained using the Douglas-Peucker algorithm are equal. In this study, especially in cases where the image edge parts are not equal, matching and classification of the obtained matches as correct and incorrect were carried out using Convolutional Neural Network (CNN).*

KEYWORDS *–2D image matching, Image processing, DELF,Classification, Convolutional Neural Network*

I. INTRODUCTION

Reassembling objects fragmented from thousands of randomly mixed pieces recovered because of archaeological excavations is an emerging problem in archaeology. It is important because it helps archaeologists access information about past cultures and civilizations [1]. The need for computer-aided methods is obvious, as reassembling such parts manually can require long, time-consuming, tedious, and precise work [2]. To reassemble archaeologically fragmented 2D objects, scanning two-dimensional historical ruin pieces and transferring them to the computer environment is considered as a method, especially for the assembly of multi-piece painted wall and ceramic fragments. In the manual assembly process,it is important to minimize the transfer of products such as oil, dirt, and acid onto historical artefact pieces, especially by minimizing handling of the parts used. With this method, more restorers and archaeologists can search for matching pieces in the computer environment in parallel, and a single restorer can glue the correct matching pieces with a single touch. Another important issue is that parts of the plaster parts can be found later during ongoing excavations. It is not possible to test a piece found years later in an area covered with plaster. This match can be easily made using old data in your computer environment, and a decision can be made to remove the plaster and replace the matching piece. To solve this problem, we have worked on 2D image fragments, image processing and neural network algorithms and methods. In this paper, the focus was on increasing the performance rate in matching edge segments of different lengths of two-dimensional parts. The DELF (DEep Local Feature) [3] and RANSAC (Random Sample Consensus) algorithm [4] was used to match edge parts of two-dimensional part images. The CNN (Convolutional Neural Network) was designed to classify the obtained matching results as correct and incorrect matching [5].

II. RELATED WORK

Problems encountered by archaeologists and restorers in reconstructing 2D archaeological remains pieces; It is the alignment between the contours of the pieces, the continuity at the border between the adjacent pieces, the continuity of color, the continuation of the thematic content, the continuation of the crack. As suggestions for solutions to these problems in the literature;Paumard, Picard and Tabia carried out their work with samples with equal and straight edges by applying cropping on the original digital image. They proposed to develop a classifier that can predict the relative position of one part relative to another. They show that solving the reassembly problem from an unordered parts list can be expressed as a shortest path problem in a carefully designed graph. To determine the positions of the pieces, they focused on the content, ignoring the boundaries of each piece.By knowing a central part in advance, the correct alignment step has been made to reassemble the image. Features of all fragments were extracted and compared to the features of the central fragment. Based on these results, it was predicted which parts were part of the image. The relative positions of the first eight segments relative to the central one was estimated. Then, the shortest path algorithm was run to reconstruct the image [6]. In this study, the pieces should consist of square edges of equal length and the middle piece should be known as the method input. Gur and Ben-Shahar used in their work rectangular pieces that can have different sizes and can be placed arbitrarily side by side along adjacent edges to solve the brick wall puzzle. In their study, it is stated that in brick wall puzzles the pieces are rectangular, have a fixed width, but can also have different heights, and the geometric installation of such pieces is more complex than square pieces that meet at the corners, and yet provides fewer constraints for reconstruction [7]. In another study, they mentioned that the outer contour features or visual cues of the part image should be used to find the best partial match in part images. Considering that their previous algorithm was poor at predicting the best-matching configuration of two neighboring parts, they approached by measuring the similarity that matches a point on the first contour with a desired target point on the second contour based on length and feature results [8]. In the works of Amigoni, Gazzani and Podico, the recombination of two pieces is the method applied to the analysis and extraction of representations of two pieces, stating that it is based on information extracted from the outlines and color contents of the pieces, without relying on any knowledge of the final image. Methods of creating an array of boundary pixels together, comparing two arrays to identify common subsequences (with satisfactory constraints on similar local curvature and color contents), and common subsequences, where the boundaries represent candidate matching parts, have been applied [9]. In this study, it is necessary to compare the edge sequence of the first piece with the edge sequences and subsequences of the second piece. As the difference between the part sizes increases, the difference between the part edge sequences increases, which increases the time complexity of the comparison process. Canyu and Xin have developedin the image recombination pipeline application, after calculating the binary match candidates from the input image parts, they classified the correct and incorrect matches from the results using the artificial intelligence algorithm convolutional neural network. To train the CNN detector, the synthesized image fragments were first aligned using the binary matching algorithm. These alignments were used to combine two image fragments. The combined images contain correct and incorrect matches. This sample set of merged images was then used to train or test the CNN. The output of the network is normalized by the softmax function, which represents the probability of correct and incorrect classification. This probability is defined as the alignment score. Those with alignment scores above a certain threshold value are defined as correctly matched images, and those below the threshold value are defined as incorrectly matched images [10]. When the synthetic input image parts used in this study are segmented along their edges, they are divided into quadrangular edge pieces of almost equal length. Therefore, the time complexity resulting from comparing edges of different sizes is not included. The study focused especially on the correct or incorrect classification of combined image samples with CNN.

III. METHODOLOGY

The proposed image fusion application consists of image acquisition, image edge segmentation, feature extraction, binary matching, and image global composition sections. Using the Python programming language and OpenCV computer vision library functions, the application parts of the proposed methods were developed on a computer with an Intel Core i7 2.30 GHz processor and 16 GB RAM memory. Digital images of objects without two-dimensional depth were used as input images. By applying digital fragmentation and rotation processes from the original archaeological as shown in Fig. 3.1 ruin images, synthetic image samples were obtained as shown in Fig.3.2. The n piece images created in the study are named img_1, img_2, ..., img_n.

Fig. 3.1. Real digital image (URL-1, 2022)

img_1 img_2 img_3

Fig. 3.2. Examples of digitally fragmented and rotated images

The images (img_1, img_2 and img_3) were firstly resized and reduced. Using the reduced images as a mask, bitwise_and operation was applied with the original image. Because we are not interested in the context and pattern of the images. As a result of the process, the images as shown in Fig.3.3 were obtained.

Fig. 3.3. Masked images

If the parts of the compared images are separated from each other, it is assumed that the part edge pixel color distributions should have the same values. The DELF (DEep Local Feature) and RANSAC (Random Sample Consensus) algorithms were used in the matching process by pairwise comparison of the images. Matched points on the images have been determined as shown in Fig. 3.4.

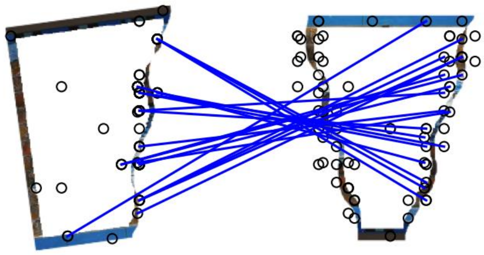

Fig. 3.4. Matched points on the images

The images were matched by keeping the first image fixed and rotating the other image according to the geometric position of the matching concentrated points close together.

Secondly, the CNN model in Fig. 3.5 was designed to classify the images obtained from previous step as correct and incorrect matching.

Input

Conv-1-1	Conv-1	Feature Extractor
Conv-1-2		
Conv-1-3		
Conv-1-4		
Max Pooling		
Conv-2-1	Conv-2	
Conv-2-2		
Conv-2-3		
Conv-2-4		
Max Pooling		
Conv-3-1	Conv-3	
Conv-3-2		
Conv-3-3		
Max Pooling		
Dense	FC-4	Classifier
Dense	FC-5	
SoftMax		

Output

Fig. 3.5. CNN Model Architecture

2919 matched images obtained from 3 different original images were used as train data. The model was trained with 130 correct labeled data and 2789 incorrect labeled data. Image data has been resized to (64, 64, 3). Epoch number was used as 10 and batch size as 32.

IV. RESULTS

To simulate distorted edges, by erasing some part of the determined image edge area as shown in Fig. 4.1, the images with matching subsections of different edge lengths were obtained. When these images were used as input in the developed application, the correct matching points were successfully detected, as seen in Fig. 4.2.

Fig. 4.1. Image with erased edges

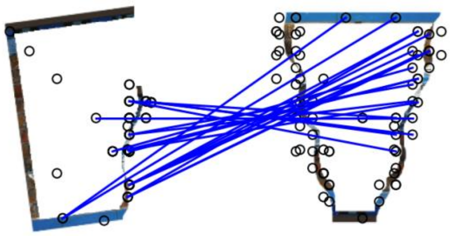

Fig. 4.2. Correctly matched points on the images

9 test matched images from 3 different original images were evaluated on the prepared CNN model as seen in Fig. 4.3. The images have been classified as correct and incorrect successfully.

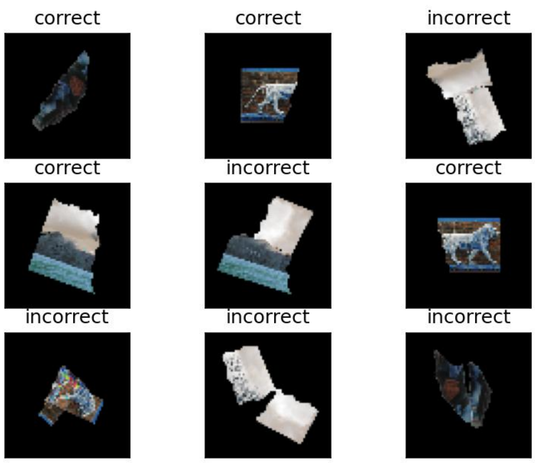

Fig. 4.3.The result of 9 test images

By using the result of CNN model, the correct matched images have been determined and matched to each other as shown in Fig. 4.4.

Fig. 4.4. Accurately matched images

V. CONCLUSION

In previous studies, instead of using the entire edge contour data to combine two residue images, the edge contour sequence was divided into pieces consisting of subsequences to speed up the matching processes, and the Canny Edge Detection algorithm was first applied to determine the object boundary pixels on the image [11]. It was observed that Polygon edge points determination process is applied on the detected outer contour. The approx Poly DP function in the Open CV image processing library, which uses the Douglas-Peucker algorithm, is commonly used to determine polygon edge points [12, 13].As a result of this approach, the images selected generally create edges of equal length. This reduces time and complexity during comparison.However, in real two-dimensional images of archaeological remains, it is known that there are distortions on the edges and edges of different lengths that break apart from each other.In this study, to take a different approach to this problem, the DELF algorithm and a CNN model were used to detect matches and classify the match results as correct or incorrect. The work can be improved by adding the feature of creating the original image from correctly matched part images.

REFERENCES

[1] V.Hristov, G.Agre,*A software system for classification of archaeologicalartefacts represented by 2D plans*, Bulgarian Acad. Sci. Cybern. Inf. Technol. 13,82–96. 2013

[2] H. C. G. Leitˇao and J. Stolfi,A Multiscale Method forthe Reassembly of Two-Dimensional Fragmented Objects,*IEEE Trans. on Pattern Analysis and Machine Intelligence,24(9):*1239–1251, 2002.

[3] H. Noh, A. Araujo, J. Sim, T. Weyand, B. Han,Large-scale image retrieval with attentive deep local features. *In: Proceedings of the IEEE International Conference on Computer Vision, pp.* 3456–3465 (2017)

[4] M.Fisher,R.Bolles,Random sample consensus: A paradigm for model fittingwith applications to image analysis and automated cartography, *Comm. of theACM 24(6)*, 381–395 (1981)

[5] A. Krizhevsky, I. Sutskever, and G. E. Hinton, Imagenet classificationwith deep convolutional neural networks, Advances *in neural informationprocessing systems*, 2012, pp. 1097–1105.

[6] MM, P., D, P., H, Tabia, Image reassembly combining deep learning and shortest path problem,*In: Proceedings of the European conference on computer vision (ECCV), pp 153–167(2018)*

[7] S. Gur, O. Ben-Shahar, From square pieces to brick walls: The next challenge in solving jigsaw puzzles, *ICCV*, (2017)

[8] M. Zhang, S. Chen, Z. Shu, S. Xin, J. Zhao, G.Jin, R. Zhang and J. Beyerer, Fast algorithm for 2D fragment assembly based on partial EMD,*The Visual Computer 32* (2016), 1–12.

[9] F. Amigoni, S. Gazzani, S. Podico,A method for reassembling fragments in image reconstruction, *Proceedings of ICIP, vols. 3,2*, 2003, pp. III-581–4.

[10] C. Le, X. Li,JigsawNet: Shredded Image Reassembly using Convolutional Neural Network and Loop-based Composition, *IEEE Transactions on Image Processing, 28(8),* 4000-4015, August 2019

[11] J. Canny , A Computational Approach to Edge Detection, *IEEE Transactions on pattern analysıs and machine intelligence,vol. pami-8, no. 6*, 1986

[12] S.T.Wu, M.R.G.Marquez,A non-self-intersection Douglas-Peucker algorithm,*In Proceedings of the 16th Brazilian Symosium on Computer Graphics and Image Processing (SIBGRAPI)*, Sao Carlos, Brazil, 12–15October 2003.

[13] D. H. Douglas,T. K.Peucker, Algorithms for the reduction of the number of points required to represent a digitized line or its caricature,*The International Journal for Geographic Information and Geovisualization, 10(2)*, 112–122.