A Texture Based Matching Approach for Automated Assembly of Puzzles

Mahmut Şamil Sağiroğlu¹, Aytül Erçil²

Sabancı University
¹msagiroglu@su.sabanciuniv.edu, ²aytulercil@sabanciuniv.edu

Abstract

The puzzle assembly problem has many application areas such as restoration and reconstruction of archeological findings, repairing of broken objects, solving jigsaw type puzzles, molecular docking problem, etc. The puzzle pieces usually include not only geometrical shape information but also visual information such as texture, color, and continuity of lines. This paper presents a new approach to the puzzle assembly problem that is based on using textural features and geometrical constraints. The texture of a band outside the border of pieces is predicted by inpainting and texture synthesis methods. Feature values are derived from these original and predicted images of pieces. An affinity measure of corresponding pieces is defined and alignment of the puzzle pieces is carried out using an FFT based image registration technique. The optimization of total affinity gives the best assembly of puzzle. Experimental results are presented on real and artificial data sets.

1. Introduction

The aim of this paper is to develop a method for the automated assembly of broken objects that have surface texture, from their pieces. The task of reassembling has great importance in the fields of anthropology, failure analysis, forensics, art restoration and reconstructive surgery. It also appears heavily in archaeology. The fact that performing reconstruction of archaeological objects from fragments manually is very time consuming motivates automatic techniques for reassembly of fragments. In general, reconstruction of objects can be regarded as a puzzle-solving problem, which contains many problems endemic to pattern recognition, computer vision, feature extraction, boundary matching, and optimization fields.

Previous works on the assembly problem have focused mainly on geometrical properties of the pieces. The puzzle pieces are represented by their boundary curves. Some approaches especially related to standard toy-store jigsaw puzzle solver use feature based matching methods. The jigsaw puzzle solving problem is a reduced and restricted version of the general assembly problem. Its computerized solution was first introduced by Freman [1], who successfully solved a 9-piece jigsaw puzzle. Other works [2,3,4] also use feature based matching approaches. These methods are relatively fast so that they manage to assemble even if the number of puzzle pieces becomes large. The main drawback of this approach is that it cannot provide detailed matching of boundaries and overlapping regions. Research involving classical jigsaw puzzle has so far ignored texture or color information to the assembly problem. There are a few approaches, which use only the color values of pixels on the boundary contour [3].

More general partial curve matching algorithms that solve the global 2D and 3D assembly problems based on geometrical properties were presented in [5,6,7]. The problem of 3D curves is addressed by [8]. The accuracy of the matching technique depends on perfect extraction of the trace of a curve and the computation of curvature and torsion. It is potentially a non-robust process and has only been tested on artificial data. Another research [9] matches 2D and 3D break curves by combining a coarse-scale representation of curves and refine iteratively via a fine-scale elastic matching. The works that achieved global assembly of pieces based on curve matching have not attempted to combine the geometrical methods with textural information.

There is great scientific interest in the archaeological community in reconstructing objects from fragments. An automatic tool that assists archaeologists in reconstructing monuments or smaller fragments would be highly beneficial. Such a tool would lead to avoiding unnecessary manual experimentation with fragile and often heavy fragments, and reduce the assembly time. Currently,
the Digital Michelangelo team is tackling the problem of assembling the Forma Urbis Romae [10]. It is a marble map of ancient Rome that has more than a thousand fragments. Their investigation is based on broken surface border curves, possibly texture patterns, and additional features of the fragments. The University of Athens has developed “The Virtual Archaeologist” [11] system, relying on the broken surface morphology to determine correct matches between fragments. This method detects candidate fractured faces, matches fragments one by one and assembles fragments into complete or partially complete entities. The Shape Lab at Brown University presents an approach to automatic estimation of mathematical models of axially symmetric pots made on a wheel [12,13]. This technique is based on matching break curves, estimated axis and profile curves, a number of features of groups of break-curves. Finally, the assembly problem is solved by maximum likelihood performance-based search. At the Technical University of Vienna, a fully automated approach to pottery reconstruction based on the fragments profile, is given [14]. Fornasier and Toniolo have developed a pattern matching algorithm for comparison of digital images by discrete Circular Harmonic expansions based on sampling theory [15]. The assumption for this method is that the photographs of the original puzzle exist.

Neglecting continuity of color and texture for adjacent fragments is a waste of valuable information for many cases. The pictorial information on a fragment consists of various components, and different specifications of surface image of pieces are dominant according to implementation field. In the archeological field, the pictorial features may include highly directional marble veining, the pattern of surface incisions, paintings on the outer and inner surfaces, carvings and horizontal circles due to finger smoothing while the pot is spinning on the wheel.

In archeology, erosion, impact damages or undesired events cause fragments to vanish or deteriorate, such as in the case of Forma Urbis Romae. This reality increases the necessity of pictorial information to solve the reconstruction of all types of puzzles, because the geometrical approaches relying on exact matching of break curves are not applicable to the assembly of the pieces, if the border of fragments has disappeared. The texture prediction method can manage to estimate possible adjacent fragments, even if there is a gap caused by erosion between two neighboring pieces.

In this paper, we design a texture prediction algorithm, which predicts the pixel values in a band outside the border of the pieces. Features obtained from the predicted texture outside a piece are correlated with original pictorial specifications of possible neighboring pairs.

The rest of this paper is organized as follows: Section 2 outlines the method used in solving the assembly problem, Section 3 presents image inpainting and texture synthesis methods that are used in predicting the expanded part of the pieces. The affinity measure used in the assembly process and the FFT technique to find the best transform are explained in Section 4. Experimental results are given in Section 5.

2. Automated puzzle assembly method

Our proposed approach is based on defining a fast and robust method that finds the best transformation of pieces that maximizes matching and continuity of textures of fragments while the geometrical constraints are satisfied. After the acquisition and preprocessing of the data, the first step is the prediction of the pixel values in a band around the border of the piece; this step is applied to all pieces separately. The prediction algorithm automatically fills in this extension region using information in the central part. The main idea in extending the picture/texture on the fragment outwards is that the correlation between the features of the predicted region and its true neighboring piece is significantly higher than alternative pairings. We use the mixture of inpainting and texture synthesis methods for prediction. Image inpainting is the process of filling in missing data in a designated part of an image or a video from the surrounding area, and texture synthesis is to create a new image with the same seed texture but of different shape to a sample region. We then derive feature values in both the original fragment and the extended region. The proposed approach does not bound the number of features or does not restrict the type of image features. Any textural feature believed to improve the success of assembly can be easily inserted into the process. A combination of the feature and confidence values is used to generate an affinity measure of corresponding pieces. The matching of pieces and achievement of the assembly is established by optimizing this affinity measure. In this paper, we propose to use FFT shift theory, which has also been used for image registration [18], to find a solution that will maximize the correlation between the predicted parts of a piece and other pieces. The main idea relies on the fact that the expanded part of the piece, whose location is sought, will be familiar to the original form of the matching piece. In this paper, we present results on 2D examples.
3. Inpainting and texture synthesis for expanding the pieces

As mentioned in Section 2, the first step in the assembly process is the expansion of each piece in a band around the border of the piece by predicting the pictorial information on the surface outwards. Inpainting and texture synthesis are two techniques that will be used to carry out this task. In this paper, we use the approach used by Criminisi [16] to predict the pixel values in a band around the border of the piece.

The source region, \( I_m^0 \), is the acquired image of the \( m^{th} \) piece. A target band, \( I_m^+ \), outwards from the \( m^{th} \) piece is defined (so \( I_m = I_m^0 + I_m^+ \)). This target band represents the extension region of the \( m^{th} \) piece. The border between \( I_m^0 \) and \( I_m^+ \) is indicated by \( \delta I_m \). This border evolves outward as the inpainting algorithm progresses. The inpainting algorithm consists of three main steps. These steps are iterated until the whole target region or band has been filled. The first step is to compute the priority, \( P \), which determines the order in which they are filled. Priority value is computed for the patches \( \Psi_p \) centered at the point \( p \) for \( p \in \delta I_m \).

Conceptually, the priority depends on continuation of strong edges, \( D \), and confidence of neighbor pixels, \( C \):

\[
P(p) = D(p) \cdot C(p)
\]

\[
C(p) = e^{\sum_{q \in \Psi_p \cap I_m^+} C(q)} \quad D(p) = \left| \mathbf{n}_p \cdot \mathbf{n}_p \right|
\]  

Where \( |\Psi_p| \) is the area of \( \Psi_p \), \( \mathbf{n}_p \) is unit vector orthogonal to the front \( \delta I_m \) at the point \( p \) and \( \perp \) indicates the orthogonal operator. This confidence value reflects the reliability of a region or a pixel, and it affects the filling order during inpainting process. Initially, we set \( C=1 \) (100\% reliability) to pixels in the original piece, and assign \( C=0 \) to the pixels in the target region to be filled. The Data term \( D(p) \) is a function of the strength of isophotes hitting the front \( \delta I_m \). This term increases the priority if an isophote flows into that patch which is important for the assembly process since it causes the linear structures to be synthesized or filled first. Therefore, the linear structures orthogonal to border of pieces are completed earlier and these points or patches get higher confidence values.

When all priorities have been computed, the highest priority, \( p^* \), is determined. The second step of the prediction process is propagating the texture and structure information into the target band. The color information is propagated via diffusion in classical inpainting techniques. In our work, as in [16], propagation of the image texture occurs by direct sampling of source region. The most similar patch for sampling is given as:

\[
\Psi'_{q} = \arg \max_{\Psi'_{p}, \Psi':q} d(\Psi'_{p}, \Psi'_{q})
\]

where \( d(\Psi'_{p}, \Psi'_{q}) \) is the distance between the already filled pixels of patches at the points \( p \) and \( q \). The patch at the point \( q \) is the most similar one and the values of each pixel to be filled in the \( p \) patch are copied directly from the patch in the \( q \) point.

The last step for iterations is to update the confidence values. After the patch \( \Psi_{p^*} \) has been filled with new values, the confidence values affected by the filling of the new patch are updated. This region is limited by the neighbors of the point \( p^* \):

\[
C(p) = C(p^*) \quad \forall p \in \psi_{p^*} \cap I_m^+
\]

As the filling proceeds, the confidence values decrease as the pixels in the predicted region get farther from the original boundary. This indicates that the color values of pixels far from border are less reliable than closer ones.

Figure 1. (a) An archeological sherd to be expanded (b) The expanded piece

4. Combining puzzle pieces

While matching or calculating similarity of possible two neighboring pieces, pixel-by-pixel comparison of two pieces is not meaningful. Thus, image features, \( (\mathbf{f}_{ik}) \), are extracted from the source, \( (\mathbf{l}_{ik}^0) \), and target regions, \( (\mathbf{l}_{ik}^+) \), for each piece after predicting the target band. Selection of the features depends on the structure of the image. Currently, only first and second moments (mean and variance of pixel intensity values) are used in the experiments. In the case of using suitable texture features, serious improvements can be obtained. The features are calculated in a window whose size depends on the resolution of the pictures on the pieces. The next step is the computation of confidence values for the features. When a feature value is extracted by using the pixels in a window, the confidence of this feature for a point depends on the confidences of all pixels in the window. Mean of all confidence of pixels in the window is assigned as confidence of the feature, \( C_m \).
Let $D_i(T_i(f_{k0}), T_i(f_{k1}))$ be the distance function between the $k^{th}$ feature values of the transformed version of $I_i^0$ and $I_i^1$ pieces. $T_i(f_{k0})$ denotes transform of the $k^{th}$ feature extracted from the $i^{th}$ piece. In our experiments, Euclidian distance is used for all features. If distances specific to texture and features of pieces are selected, the performance of assembly might improve.

$$F_{cost} = \sum_i \sum_k \sum_j w_k D_i(T_i(f_{k0}), T_j(f_{k1}))-1 C_j' C_i'$$

(5)

(See [17] for details). Total cost is the summation of similarity for all points in space. This value goes towards negative if there exists a good matching between the pictures on the candidate pieces. The only parameter of this performance measure that represents the goodness of the assembly of pieces based on textural features is the transformation of pieces, $T$. The fitness between the pieces is increasing while the Cost function is being optimized.

Although, the puzzle assembly problem can be stated as the optimization of the above cost function, the optimization problem is too computationally costly. For all practical purposes, the minimizing the above distance function, $D$, is equivalent to maximizing the correlation between the pieces. We will therefore use the FFT shift theory to find a solution that will maximize the correlation between the predicted parts of a piece and other pieces. Let us first consider the solution to a 2 pieces puzzle. The solution set consists of the piece $I_0^0$ and the transformed version of the piece $I_1^0$. The transformation consists of translation and rotation ($T_i=\Delta x_i, \Delta y_i, \Delta \theta_j$). The transformation that gives maximum correlation between $I_1$ and $I_0$ is the best match between the two pieces and hence is the solution defined as below:

$$S^2_{general} = \argmax_{T_i} \sum_{k} C(f_{k0}, T_i(f_{k1}))$$

(6)

$C$ denotes the correlation operator. The maximum correlation solution does not guarantee the real solution to the puzzle since it does not incorporate the physical constraint that two pieces cannot overlap. This constraint can be expressed in terms of correlations:

$$(I_0^0) \cap (T_i(I_1^0)) = \emptyset \quad \text{or} \quad C(I_0^0, T_i(I_1^0)) = 0$$

(7)

Hence the real solution set is given by:

$$S^2 = \{ \argmax_{T_i} \sum_{k} C(f_{k0}, T_i(f_{k1})) \left| C(I_0^0, T_i(I_1^0)) = 0 \right. \}$$

(8)

The solution set lies where the original pieces have correlation 0, and the $I_1^{th}$ piece has maximum correlation with the $I_0^{th}$ piece.

These correlations can be carried out very fast using FFT operations as in image registration methods [15].

$$S^2_{general} = \argmax_{T_i} \sum_{k} C(f_{k0}, T_i(f_{k1}))$$

(9)

and also

$$C(I_0^0, I_1^0) = \tilde{F}(F(I_0^0), F^*(I_1^0))$$

(10)

where $\argim$ returns the indices of the max value. If we substitute (9) and (10) in (8);

$$S^2 = \argmax_{T_i} \sum_{k} \left[ F(f_{k0}), F^*(f_{k1}) \right] L[\tilde{F}(F(I_0^0), F^*(I_1^0))]$$

where

$$L[x] = \begin{cases} 0 & \text{if } x \neq 0 \\ 1 & \text{if } x = 0 \end{cases}$$

(11)

$F$, $F^*$ and $\tilde{F}$ denote the Fourier operator, its complex conjugate and the inverse Fourier operator, respectively. When the inverse Fourier value in the second part of the formula is zero, the maximum values of the first part of the formula gives the ideal transformation parameters.

Figure 2 shows the steps of a 2 pieces puzzle assembly. In Figure 2.a, there are two original pieces. In Figure 2.b, the expanded pieces are shown. Then, the features are derived from the expanded pieces. In 2.c, only the mean feature is shown. The image in 2.d represents the correlation matrix between these two expanded pieces. This is the general solution for the problem. If the constraint expression is applied, however, the image in 2.e is found. This matrix is the output of the inner part of expression (11). The arrow in Figure 2.e indicates the maximum point in this matrix. The indices of this maximum point give the translation coefficients. The last image in Figure 2 shows the solution if this translation is applied to the second piece.

The FFT solution helps to solve translation only. The rotation can be solved using polar coordinates in an iterative scheme by using the same formulation. (See [18] for details).

The solution outlined above can be generalized to the solution of more realistic puzzles with larger number of pieces. Let $n_p$ be the number of pieces involved. The solution is the set of appropriate transformations of each piece:

$$S^{n_p} = \left\{ 0, T_1, T_2, T_3, \ldots \ldots, T_{n_p} \right\}$$

(12)

Assume that all the pieces are randomly dispersed on a big enough board ($B$). We randomly select a piece ($I_i$). For this piece, the transformation giving maximum correlation is obtained using the above technique.
In these situations, the affinity measure developed above is used (Equation (5)). For a possible placement given by the proposed technique, the $F_{cost}$ is calculated. If the algorithm reaches to a new solution that has a lower cost value than before, the new one becomes the best placement. These iterations continue until the last $N$ possible placement cannot offer a better cost. This $N$ value directly depends on the complexity of the puzzle. The main argument of the complexity for a puzzle is the number of the pieces. So the number $N$ mainly depends on the $n_p$ value. We assume $N \approx n_p^2$ in our experiments.

Hence the final version of the algorithm has the following two additional steps:

6. Find $F_{cost}$ and if the new one is better then before, save the transformation as the best solution and clear the counter $N$.
7. Go to step 1, until $N$ reaches to $n_p^2$

5. Experimental results

We will demonstrate the results of the proposed algorithm on three different datasets. The first dataset (13 pieces) from Stanford university website is part of the Forma Urbis Romae dataset [10] which is a marble map of ancient Rome that has more than a thousand fragments. For this experiment, the image of a fragment from this dataset is broken artificially. Figure 3.a shows the pieces in the dataset and Figure 3.b shows the final assembly obtained.

The second dataset consists of 21 pieces of a original ceramic tile. Figure 4.a shows the pieces to be assembled. Figures 4.b–d give three possible solutions. The corresponding cost functions are also given. It is noted that all the solutions are visually feasible solutions in terms of the texture and geometry information and the correct solution has the minimum cost function.

The last experiment has pieces from two different ceramic tiles. This experiment is important since in a real archaeological set-up, pieces may come from two
or more objects. 10 pieces of the tile used in Experiment 2 are mixed with 9 pieces from another tile. Resulting assembly is given in Figure 5.

![Figure 4](image1)

(a) $F_{cost} = 0$  
(b) $F_{cost} = -18577$

(c) $F_{cost} = -17841$  
(d) $F_{cost} = -20250$

Figure 4. (a) Initial layout of the pieces (b,c,d) $F_{cost}$ for 3 possible solutions.

![Figure 5](image2)

Solution obtained for mixture of pieces from two different tiles.

6. Summary and conclusions

We presented a method for the automated puzzle assembly problem using surface texture and picture. The approach is based on expanding the boundary of each piece using inpainting and texture synthesis methods and maximizing the correlation based on matching feature values obtained from these predicted regions. Initial experiments show that this approach is very promising for the automated puzzle assembly problem. Future work will concentrate on upgrading the search tool and generalizing the presented algorithm to solving 3D puzzles.

7. References