



HAL
open science

Texture Based Approach to Puzzle Assembly

Mahmut Samil Sagirolu, Aytul Ercil, Can Ozmen, Selim Balcisoy

► **To cite this version:**

Mahmut Samil Sagirolu, Aytul Ercil, Can Ozmen, Selim Balcisoy. Texture Based Approach to Puzzle Assembly. Virtual Retrospect 2005, Robert Vergnieux, Nov 2005, Biarritz, France. pp.73-79. hal-01759626

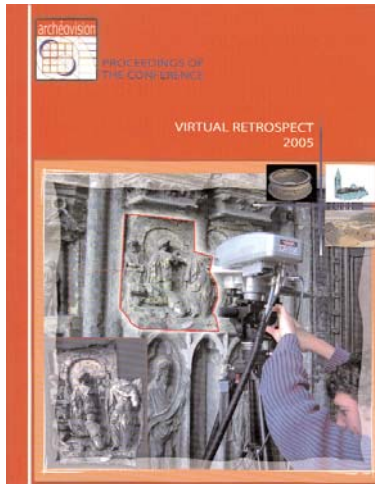
HAL Id: hal-01759626

<https://hal.science/hal-01759626>

Submitted on 5 Apr 2018

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Vergniew R. et Delevoie C., éd. (2006),
Actes du Colloque Virtual Retrospect 2005,
Archéovision 2, Editions Ausonius, Bordeaux

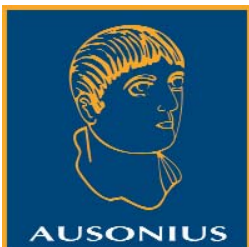
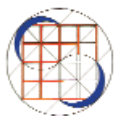
Tiré-à-part des Actes du colloque Virtual Retrospect 2005

Biarriz (France) 8, 9 et 10 novembre 2005



M.S. Sagiroglu, A. Erçil, C. Özmen, S. Balcisoy

Texture Based Approach to Puzzle Assembly p.73-79



Conditions d'utilisation :
l'utilisation du contenu de ces pages est limitée à un usage
personnel et non commercial.
Tout autre utilisation est soumise à une autorisation préalable.
Contact : virtual.retrospect@archeovision.cnrs.fr

<http://archeovision.cnrs.fr>



TEXTURE BASED APPROACH TO PUZZLE ASSEMBLY

Mahmut Şamil Sağıroğlu
Aytül Erçil
Can Özmen
Selim Balcısoy

Sabancı University

msagiroglu@su.sabanciuniv.edu
aytulercil@sabanciuniv.edu
canozmen@su.sabanciuniv.edu
balcisosy@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.. 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. The confidence of this process is also defined. Feature values are derived from these original and predicted images of pieces. A combination of the feature and confidence values is used to generate an affinity measure of corresponding pieces. The optimization of total affinity gives the best assembly of the puzzle.

Experimental results are presented as real and artificial data.

An application, V-Stitch is also developed, with the purpose of providing a real-time virtual reality software system with an intuitive interface for using semi-automatic algorithms developed for the arrangement of archeological sherds.

Keywords : archeological reconstruction — partial matching — puzzle problem — inpainting — texture synthesis.

1 Introduction

In archaeological sites, we may encounter a large number of irregular fragments resulting from one or several broken objects. The reconstruction of the original object is a tedious and laborious task. In this paper, we consider the complex problem of automatically assembling 2D/3D objects, from their fragments, using input from multiple cameras or a 3-D scanning system with synchronized texture facility.

Previous works on the assembly problem have focused mainly on the geometrical properties of the pieces. The puzzle pieces are represented by their boundary curves. As the fractions of boundaries are adjacent and thus similar, a pairwise affinity

measure is computed by partial curve matching. Some approaches especially related to standard toy-store jigsaw puzzle solvers use feature based matching methods. The problem of jigsaw puzzle solving is a reduced and restricted version of the general assembly problem. Its computerized solution was first introduced by Freeman [1], who successfully solved a 9-piece jigsaw puzzle. Other works [2, 3, 4, 5] 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 they cannot provide detailed matching of boundaries and overlapping regions. Research involving classical jigsaw puzzles has so far ignored texture or color information in the assembly problem. There are a few approaches, which use only the color values of pixels on the boundary contour [4].

More general partial curve matching algorithms that solve the global 2D and 3D assembly problems based on geometrical properties were presented in [6, 7, 8]. The problem of 3D curves is addressed by [9]. The accuracy of the matching technique depends on the perfect extraction of the trace of a curve and the computation of the curvature and torsion. It is potentially a non-robust process and has only been tested on artificial data. Another research paper [10] matches 2D and 3D break curves by combining a coarse-scale representation of curves, refined iteratively via a fine-scale elastic matching. The works that achieved, a 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 archeologists 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 [11]. 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” [12] 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 the automatic estimation of mathematical models of axially symmetric pots made on a wheel [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 a maximum likelihood performance-based search. At the Technical University of Vienna, a fully automated approach to pottery reconstruction based on the of fragments profile, is given.[14] Neglecting continuity of color and texture for adjacent fragments is a waste of valuable information in many cases. The pictorial information on a fragment consists of various components, and different specifications of surface image of pieces are dominant according to the implementation field. In the classical jigsaw puzzles, the essentials of assembly depend on the alignments of object edges (e.g. picture of a house), the similarity of colors (e.g. cloud drawing) and the continuity of textural properties (e.g. grass of a garden) for the adjacent pieces. In the archeological field, the pictorial features may include highly directional marble veining, the pattern of surface incisions, painting 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. 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 have 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 shall design a texture prediction algorithm, which predicts the pixel values in a band outside the border of the pieces with a confidence measure. Features obtained from the predicted texture outside a piece are correlated with original pictorial specifications of possible neighboring pairs. An affinity measure of corresponding pieces that utilizes all kind of image information, such as continuity of edges, textural patterns, and color similarities is defined and the assembly problem is stated as the optimization of this affinity measure.

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 cost function/affinity measure used in the assembly process is explained in Section 4. Experimental results are given in Section 5. The final Section 6 presents a Virtual Reality application for a natural interaction of users with the system.

2 Automated puzzle assembly method

Our proposed approach is based on defining a performance measure that represents the appropriateness of the assembly based on textural features and geometrical shape, and to find best transformations of pieces that maximize the matching of textures of fragments while the geometrical constraints are being satisfied. After the collection of visual 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. While extending the fragment image, we introduce confidence of extension as a new parameter in the prediction phase of the assembly problem. This parameter represents the reliability of extended values and will be used by later processes. The confidence depends on the structure of the texture such as continuity of edges, roughness of texture and distance to the border of the original fragment. We then derive feature values from 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. The next step is to determine a similarity or a cost function between two textural regions. The final goal of the proposed approach is to establish an affinity measure of corresponding pieces by the combination of the feature and confidence values. The matching of pieces and achievement of the assembly is established by optimizing this affinity measure. To improve the assembly, we have to be able to sense whether a particular arrangement of pieces improves the puzzle or not; this is done using a total affinity measure defined as the sum of affinity measures of all points in the space. The space may be 2D such as for the broken marble problem or 3D such as for the pot assembly problem. In this paper, we shall present results on 2D examples. The extension of the proposed method to 3D is computationally costlier, but is theoretically possible.

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 into 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.

Image inpainting refers to the process of filling-in the missing areas. It is usually applied to the task of restoring photographs, films or paintings, and the removal of occlusions, such as subtitles, stamps and text. In [15, 16], a series of partial differential equations is used to mathematically model this process. These techniques determine how the linear structures (called isophotes) propagate into the region to be inpainted. Other inpainting approaches are the Total variational (TV) and Curvature-drive diffusion models (CCD)[17]. TV uses an Euler-lagrange equation to minimize total variation and employs anisotropic diffusion. Such a method handles noise well, but does not complete broken edges. CCD is based on the TV algorithm and geometric information of isophotes. The drawback of these methods is the blurring of the inpainted image introduced by the diffusion process in the larger filling regions.

Texture synthesis is an active research topic in computer vision, which has broad applications such as foreground removal, lossy image compression, and texture generation.

The problem of texture synthesis is to fill large image regions with a sample texture. This method, which replicates consistent textures, can be used in the extension of images, but it has problems to fill in real image patterns. Linear structures such as a drawing of a line or crossing regions of different textures usually include high frequency components, which prevent the generation of natural images by this approach.

To overcome the drawbacks of inpainting and texture synthesis algorithms, the method presented in [BVS03] first decomposes the image into the sum of two components with different basic characteristics and then reconstructs each one of these components separately with inpainting and texture synthesis. Another approach by Harrison [18] and Criminisi [19] use exemplar-based synthesis for the object removal process.

In this paper, we use the approach used by Criminisi to predict the pixel values in a band around the border of the piece, however, the implementation is slightly different.

The source region, Φ_i , is the acquired image of the i^{th} piece. A target band, Ω_i , outwards from the i^{th} piece is defined. This target band represents the extension region of the i^{th} piece. The border between Φ_i and Ω_i is indicated by $\delta\Omega_i$. 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. The priority value is computed for the patches Ψ_p centered at the point $p \in \Omega_i$. Conceptually, the priority depends on the continuation of strong edges, D , and confidence of neighbor pixels, C :

$$P(p) = C(p) \cdot D(p) \quad (1)$$

$$C(p) = \frac{\sum_{q \in \Psi_p \cap \Phi_i} C(q)}{|\Psi_p|}, \quad D(p) = \left| \nabla I_p^\perp \cdot n_p \right| \quad (2)$$

where $|\Psi_p|$ is the area of Ψ_p , n_p is unit vector orthogonal to the front $\delta\Omega$ at the point p and \perp indicates the orthogonal operator. This confidence value reflects the reliability of a region or a pixel, and it effects the filling order during the 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 $\delta\Omega$. 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 the 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 [CPT03], propagation of the image texture occurs by the direct sampling of the source region. The most similar patch for sampling is given as :

$$\Psi_{q'} = \arg \min_{\Psi_q \in \Phi_i} d(\Psi_{p'}, \Psi_q) \quad (3)$$

where $(\Psi_{p'}, \Psi_{q'})$ 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 $\{\text{neighbor } p' \mid \text{neighbor } p' \in (\Psi_{p'} \cap \Omega_i)\}$ 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' .

$$C(p) = C(p') \quad \forall p \in \Psi_{p'} \cap \Omega_i \quad (4)$$

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 the border are less reliable than closer ones.

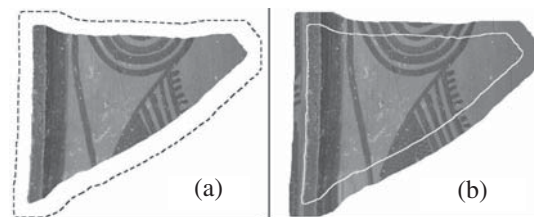


Fig. 1: (a) An archaeological sherd to be expanded
(b) The expanded piece.

4 Combining puzzle pieces

While matching or calculating the similarity of possible two neighboring pieces, pixel-by-pixel comparison of two pieces is not meaningful. Thus, image features, (f_i^k) , are extracted from the source and target regions for each piece after predicting the target band. The selection of the features depends on the structure of the image. Currently, only first and second

moments (mean and variance) 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. The mean of all confidence of pixels in the window is assigned as confidence of the feature, C_i^j .

Let $D^k(f_i^k(T_i(x,y,\theta)), f_j^k(T_j(x,y,\theta)))$ be the distance function between the k^{th} feature values of the i and j pieces. $T_i(x,y,\theta)$ denotes the transform of the i^{th} piece at the point (x,y,θ) . 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. For the simplicity of expressions, the $T_i(x,y,\theta)$ parameter for each variable will not be shown.

We set a threshold, Th^k , for the k^{th} feature distance, so that the more similar the pieces are, the larger the negative value the similarity measures, S^k , will take or visa versa.

$$S^k = D^k(f_i^k, f_j^k) - Th^k \quad (5)$$

$$\sum_k^{n_k} S^k = \sum_k^{n_k} [w^k D^k(f_i^k, f_j^k)] - 1 \quad (6)$$

where n_k is the number of features. $\sum_k^{n_k} S^k$ gives the total

similarity between the i^{th} and the j^{th} pieces at the point (x,y,θ) .

We can transform $\sum_k^{n_k} S^k$ into (6) by dividing all S^k into Th^k

and normalizing the total constants to 1, so that both of them give related responses for the same inputs. w^k are the weight values for the k^{th} feature and are inversely proportional to Th^k .

$$\sum_j^{n_p} \left[\sum_k^{n_k} w^k D^k(f_i^k, f_j^k) - 1 \right] C_j^i \quad (7)$$

where n_p is the number of pieces in the puzzle. Expression (7) denotes that total similarity between the i^{th} and the j^{th} pieces are weighted according to the j^{th} confidence values, since low confidence points are unreliable, even if two pieces are similar.

After weighting the similarities, the summation for all j pieces where i is different than j shows how much the i^{th} piece fits the other pieces at $T_i(x,y,\theta)$.

$$m_1(x, y) = \frac{\sum_i^{n_p} \sum_{j=i+1}^{n_p} \left[\sum_k^{n_k} w^k D^k(f_i^k, f_j^k) - 1 \right] C_j^i C_i^j}{\sum_i^{n_p} C_i^j} \quad (8)$$

This is the first part of the Cost or Affinity function and is derived from the weighted mean of (7). It is the summation of similarities for possible pairs. This value goes towards negative if there exists a good match between the pictures on the candidate pieces.

$$m_2(x, y) = \sum_i^{n_p} \sum_{j=i+1}^{n_p} w_c L(C_i) L(C_j), \quad L(x) = \begin{cases} 1 & x = 1 \\ 0 & x \neq 1 \end{cases} \quad (9)$$

The second part of the function is for embedding the geometrical constraints to Cost or Affinity. In reality, two pieces cannot overlap at any point. The confidence values are used to formulize overlapping operation. The L function will be 1 only for pixels in the original part of the image, otherwise it will be 0. Using a sufficiently large w_c , the Cost increases when the original parts of the i^{th} and j^{th} images overlap.

$$F_{\text{cost}} = \sum (m_1 + m_2) \quad (10)$$

Total cost is the summation of similarity and geometrical constraints terms for all points in space. The only parameter of this performance measure that represents the exactitude of the assembly of pieces based on textural features and geometrical shape is the transformation of pieces, T_i .

Two types of optimization methods are used in the experiments. The first one depends on the best replacement strategy. Initially, the transformation of pieces are randomly assigned. The algorithm progresses by finding the best movement in each step. When the function is stuck into a local minimum, two randomly selected pieces are exchanged.

All local minima are buffered to find the best assembly. The algorithm is stopped if the function reaches the best value in the local minima buffer more than n times.

The second method depends on the pairing of the pieces. Initially, the algorithm searches for the best pair that gives the minimum cost. Then, these paired pieces are merged to produce a unique piece. The algorithm is stopped when all the pieces in the puzzle are combined and become one piece. In this method, the algorithm backtracks when the pairing cannot improve the cost. To implement this method, the confidence

$$C_{\text{new}}^i = 1 - \prod_{i \in M} (1 - C_i^j) \quad (11)$$

and feature values of the new piece should be defined after the merging process.

$$f_{\text{new}}^k = \frac{\sum_{i \in M} \left[\prod_{j \in M, i \neq j} (1 - C_j^i) \cdot C_i^j \cdot f_i^k \right]}{\sum_{i \in M} \left[\prod_{j \in M, i \neq j} (1 - C_j^i) \cdot C_i^j \right]} \quad (12)$$

M is the set of pieces that will be merged. (11) gives the new confidence value for overlapping points of pieces. It expresses that new the confidence value is equal to 1 if one of the pieces has a confidence of 1, otherwise it is the geometrical mean of possible confidence values at that point. (12) gives the new k^{th} feature values by calculating the weighted mean of pieces in the set M .

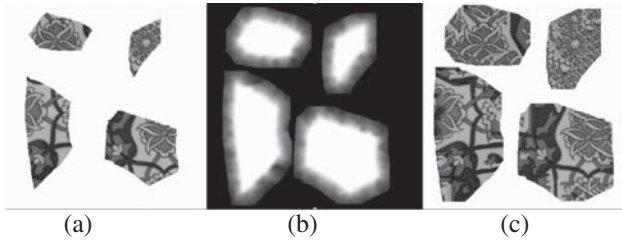


Fig. 2: (a) A puzzle consisting of 4 pieces, (b) confidence values of the predicted regions (c) expanded versions of the pieces. ($F_{\text{cost}}=0$).

5 Experimental results

The behavior of the defined affinity measure is observed under different scenarios. The first one is whether the edges continue on the neighboring pieces or not. In the inpainting phase, the edges obtain higher confidence values as was explained earlier. The higher confidence values force the cost function to locate the pieces properly. The second important criterion is similarity of corresponding textures on the neighboring pieces. The distance measure in the cost function attracts similar textures together if the expanded regions of the pieces are accurately inpainted.

In this paper, we present results from two different datasets. The first dataset consists of 21 pieces of a ceramica tile. We will give the details of the experiment with 4 pieces so that the details of the images can be distinguished. The second dataset (13 pieces) from the Stanford university website is part of the Forma Urbis Romae dataset[L00] which is a marble map of ancient Rome that has more than a thousand fragments.

In figure 2, the original images, confidence images and expanded images of 4 pieces are placed, respectively. The cost in the solution space is equal to zero for this placement, because the expanded or original regions of the 4 pieces are not overlapped anywhere. In figures 3 and 4, different assembly stages and the corresponding cost values are shown. Two neighboring pieces are placed closer with a shift in figure 3a, and their corrected placement is represented in figure 3b. The main difference between the cost values of (a) and (b) is that the edges don't continue in (a) although the neighboring textures are mostly similar. In figure 3c, the third piece is placed in its right position, but the original (real) regions of the fourth piece and the third piece are overlapped; in other words, the fourth piece violates the geometrical constraints. For this situation, the second part of the cost function (m_2) becomes dominant and the cost increases seriously. In figure 3d, we see the fourth piece is placed in the most appropriate location.

Figure 4 shows the completed reconstruction with the associated cost. Figure 5 shows the steps of assembling the 13 pieces from the Forma Urbis Romae dataset.

The optimization program developed is also tested against erosion and missing pieces. Even if the edges of the pieces were eroded or one of the puzzle pieces had disappeared, the program was able to find the right assembly for the puzzles under test.

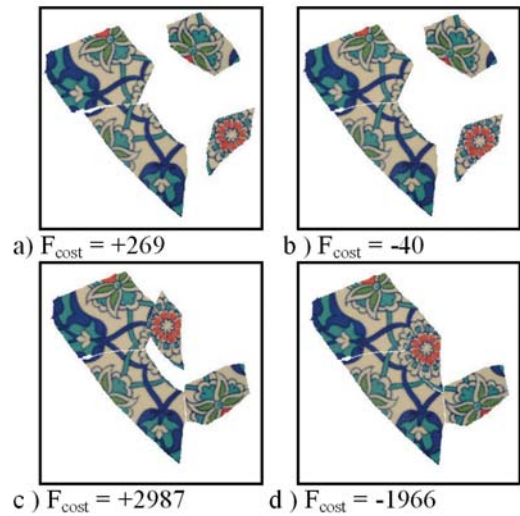


Fig. 3: (a), (b), (c) Total cost for different layouts (d) Total cost for the completed puzzle.



Fig. 4: Total cost for the completed puzzle $F_{\text{cost}}=-20,076$.

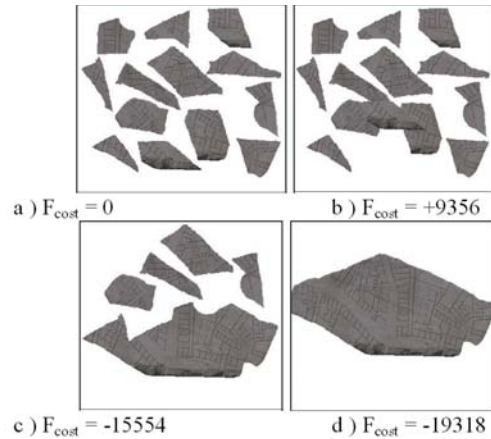


Fig. 5: (a), (b), (c) Total cost for different layouts (d) Total cost for the completed puzzle.

6 Virtual Reality System

Cultural heritage researchers would like to perform fitting experiments with the scanned 3D objects. As expert users they can intervene and suggest solutions where the automatic techniques fail. On the other hand the user may select two pieces and the system would compute their fitting likelihood. The user can decide upon this information. We propose that the combination of human expertise and pure software solutions will lead to better results than manual or automatic only solutions.

We developed an application, V-Stitch, with the purpose of providing a real-time virtual reality software system with an intuitive interface for using semi-automatic algorithms developed for the arrangement of archeological sherds. The system loads the 3D models of sherds together with a semantic information file, which includes the fitting metric of two sherds computed by the matching algorithm presented in section 4. Thus the system provides pre-computed pair-wise matches between sherds as suggestions to the user.

We considered several input and output technologies in order to achieve optimum results in accordance with the constraints inherent in the application domain. The considered input devices are as follows :

Device	Description	Sensor	Accuracy/Rate
3Dconnexion Spaceball® 5000	3D controller with 6-DOF (position & orientation) and 11 buttons	Optical	Adjustable
Intersense InertiaCube2	360 degrees orientation (3-DOF) tracker	Gyroscopic	Adjustable (0-1200 degrees per second) with 180 Hz update rate.
Polhemus Fastrack	Motion tracker with 6-DOF (position & orientation)	Electromag.	0.03 inches for position, 0,05 degrees for orientation (dependant upon sensor/source distance) with 120 Hz - 30 Hz rate (dependant upon number of sensors)

We found that the Polhemus Fastrack tracker was unsuitable for our application because of space constraints as well as the sensitivity of the equipment to environmental conditions such as the presence of conductive metals. The Intersense InertiaCube2 tracker was similarly found lacking in its usability as a controller device because of its design. We selected the 3Dconnexion SpaceBall® as the preferred input device because of its suitability to manipulate 3D objects. Its small space requirements and adjustable update rate also made it a good choice for our application.

The considered output devices are as follows :

Device	Description	Sensor	Resolution
Prosystems Pro View 3D Mobile	3D Passive Stereo Projector	Linear polarized filters and glasses	1280 x 1024
Trivisio ARvision-3D HMD	Stereo head mounted display	Two SVGA microdisplays	800 x 600

We decided that the 3D projector was inappropriate for our purposes because of its high price and space constraints although it has the higher resolution. The ARvision-3D head mounted display was selected as the preferred output device because of its small size and suitability for desktop use.

We were able to achieve a suitable frame-rate for real-time use with our resulting application on a desktop computer with a Intel Pentium 4 – 3 GHz processor and a ATI Radeon X800XT graphics coprocessor.



Fig. 6: Snap-shots from the VR system where the user tests two sherds.

7 Summary and conclusions

We have 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 and minimizing a cost function 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 optimizing the search for best transformation and generalizing the presented algorithm to solving 3D puzzles.

Bibliography

- [1] Freeman, H. and L. Garder (1964): “A pictorial Jigsaw puzzles: The computer solution of a problem in pattern recognition”, IEEE Trans. Electron. Comput. 13, 118-127.
- [2] Goldberg, D., C., Malon and M. Bern (2002): “A global approach to automatic solution of jigsaw puzzles”, In Proc. of Conf. on Computational Geometry, 82-87.
- [3] Wolfson, H., E. Schonberg, A. Kalvin and Y. Lambdan (1988): “Solving jigsaw puzzles using computer vision”, Annals of Operations Research, 12, 51-64.

- [4] Chung, M.G., M.M. Fleck and D.D. Forsyth (1998): "Jigsaw Puzzle Solver Using Shape and Color", in Proc. of ICSP.
- [5] Kosiba, A.D., P.M. Devaux, S. Balasubramanian, T.L. Gandhi and R. Kasturi (2001): "An automatic jigsaw puzzle solver", in Proc. Int. Conf. Pattern Recognition .
- [6] Kong, W.and B.B. Kimia (2001): "On solving 2D and 3D puzzles using curve matching", in Proc. of CVPR, Hawaii, USA.
- [7] Radack, G.M. and N.I. Badler (1982): "Jigsaw puzzle matching using a boundary-centered polar encoding", CGIP, 19, 1-17.
- [8] Wolfson, H.(1990): "On curve matching", PAMI, 12, 483-489.
- [9] Ucoluk, G.and I.H. Toroslu (1999): "Automatic reconstruction of broken 3-D surface objects", Comp. and Graph., 23, 573-582.
- [10] Stol, J.and H. Leitao (2002): "A multiscale method for the reassembly of two-dimensional fragmented objects", PAMI .
- [11] Levoy, M. (2000): "The digital michelangelo project:3D scanning of large statues", Computer Graphics, in Proc. SIGGRAPH 2000, New York, 131-144.
- [12] Papaioannou, G., E.-A. Karabassi and T. Theoharis (2001): "Virtual archaeologist: assembling the past", IEEE Computer Graphics and Applications, 21(2), 53- 59.
- [13] Willis, A. and D. Cooper (2003): "Accurately estimating sherd 3D surface geometry with application to pot reconstruction", In CVPR Workshop, ACVA, (June).
- [14] Kampel, M. and R. Sabalatnig .(2003): "Profile based Pottery Reconstruction", in Martin D. (ed). IEEE/CVPR "Workshop on Applications of Computer Vision in Archaeology", CD-ROM, Madison, USA.
- [15] Bertalmio, M., G. Sapiro, V. Caselles, C. Ballester (2000): "Image inpainting", in proc.SIGGRAPH 2000, New Orleans, LU, 417-424.
- [16] Bertalmio, M., L. Vese, G. Sapiro, M. Oliveria, B. Bowen, R. McKenna and Y. Chang (2001): "Fast digital image Inpainting", Proc. of International Conference on Visualization, Imaging and Image Processing, Marbella, Spain, 261-266
- [17] Chan T., and J. Shen (2000): "Non-texture inpainting by curvature-driven diffusions" (CCD), UCLA CAM TR 00-35, 266; Bertalmio, M., L. Vese, G. Sapiro and S. Osher: (2003): "Simultaneous structure and texture image inpainting", in Proc. Conf. Comp. Vision Pattern Rec., Madison, WI.
- [18] Harrison, P. (2001): "A non-hierarchical procedure for re-synthesis of complex texture", in Proc. Int. Conf. Central Europe Comp. Graphics, Visua. And Comp. Vision, Plzen, Czech Republic.
- [19] Criminisi, A., P. Perez and K. Toyama (2003): "Object removal by exemplar-based inpainting", CVPR, 721-728.

