Toward A Framework for Assembling Broken Pottery Vessels

Stuart Andrews and David H. Laidlaw
Department of Computer Science, Brown University
Providence, Rhode Island 02912 {stu,dhl}@cs.brown.edu

This paper addresses how to automatically reconstruct pottery vessels from a collection of sherds using a variety of features and their comparisons. To solve the problem, we designed a computational framework that is founded on the primitive operations of "match" proposal and evaluation. A match defines the geometric relationship between a pair of sherds. This framework affords a natural decomposition of the computation required by an automatic assembly process and provides a concrete basis to evaluate the utility of different features and feature comparisons for assembly. Pair-wise matches are proposed and subsequently evaluated by a series of independent feature similarity modules. Assembly strategies are abstracted from the feature-specific sherd details and operate solely in terms of the probabilistic output of pair-wise proposals and evaluations.

Our framework, which is modular and extensible, paves the way for a system to automatically reconstruct pottery vessels. We demonstrate a greedy assembly strategy that predicts likely pairs and triples of sherds using a handful of proposal and evaluation modules.

Previous attempts to automate the task of reconstructing pottery vessels have relied on a single feature, and sometimes user intervention, to direct the search (Ucoluk & Toroslu 1999), (Papaioannou, Karabassi, & Theoharis 2001), (da Gama Leito & Stolfi 1998). While (Cooper *et al.* 2001) accounts for more than one feature using complex parametric models, we propose a conceptually simpler and modular system for integration akin to (Pankanti, Jain, & Tuceryan 1994) and (Keim *et al.* 1999; Keim, Shazeer, & Littman 1999). Our framework and assembly strategy are similar to (Jepson & Mann 1999) where they search for a plausible scene interpretation.

Methods

Our framework is centered around pair-wise comparison of sherds. It performs two types of comparisons: proposals, which propose a "match" or relative placement of the two sherds; and evaluations, which evaluate the likelihood of a given relative placement. These primitive operations can be realized using many different geometric features – we demonstrate only a few. Assembly modules build on top of these primitive operations.

Proposal: Given an input pair of sherds, a proposal module generates a list of matches for the pair. The example we implemented generates a list of matches where each has a corner from each sherd coincident and one adjacent edge aligned.

Evaluation: An evaluative module produces a $\mathcal{X}^2$ statistic which - as a sum of independent and normally distributed squared residuals - provides a well-known way to evaluate the match likelihood and to form a joint likelihood from an ensemble of evaluative modules (Press *et al.* 1986). We implemented four evaluative modules. Assuming rotational symmetry of the vessels, the first two modules measure individual residuals using the axis of rotation for the respective sherds:

$\mathbf{E_1}$ Alignment of the axes: the residual is the angle between the axes.

$\mathbf{E_2}$ Overlap of the axes: the residual is the perpendicular distance between the axes.

The next two modules measure the alignment of the inside and outside break curves of the sherds. These curves delineate the break on the inside and outside surface of the vessel. The residuals that comprise the $\mathcal{X}^2$ are measured at closest-point pairs. For a given point A_i on one sherd's break curve, the closest-point pair is formed by the closest point $\tilde{A}_i$ on the other sherd's break curve. Points are sampled at intervals coarse enough to justify the independence of residuals.

$\mathbf{E_3}$ Distance between break curves: the residuals are $\| A_i - \tilde{A}_i \|$.

$\mathbf{E_4}$ Alignment of tangent vectors: the residuals are defined by the angle between tangent vectors to the break curves at A_i and $\tilde{A}_i$.

Assembly: Our prototype assembly algorithm uses the modules defined above to search for the original correct configuration of sherds. Our strategy starts by generating a set of match candidates for each pair using the proposal module. Then, we adjust the relative placement of each candidate match to maximize the ensemble likelihood and thereby improve the alignment of all features. This step is performed using a quasi-Newton algorithm for continuous function optimization (NAG 1993). The resulting optimized

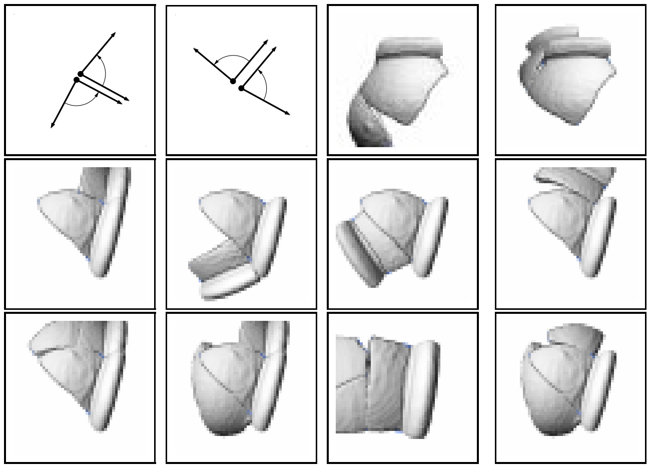

Figure 1: (by row) match proposals, ranked matches and best triples.

matches are then ranked according to their maximized ensemble likelihood values. Finally, we use a greedy strategy to select pair-wise matches to form triples. These are optimized in a similar fashion as the pairs by defining a three-way match likelihood as the sum of the three measurable pair-wise match likelihoods.

Results

The following section documents the results of our assembly algorithm when applied to a subset of 8 neighboring sherds from a 16 sherd test vessel. Of the 28 possible pairs, there are only 13 valid pair-wise matches in the correct reconstruction. The first row of the Figure depicts the geometry of two corner alignments and two examples of proposed matches. Next, we show the top four optimized matches for one pair of sherds from the vessel rim ranked by decreasing ensemble likelihood [0.999, 0.863, 0.574, 0.548]. For this pair, the ensemble likelihood identifies the correct match. In the same way, this procedure correctly identifies 6 out of the 13 valid matches. Nine correct matches are ranked among the top three matches per pair. In the last row, we show the most likely triples found by merging pairs and evaluating according to the 3-way likelihood described above; only the last one is incorrect.

Discussion and Conclusions

Our framework paves the way for a completely automated reconstruction system. There still remains several unresolved issues such as how to automate feature segmentation and how to deal with missing or faulty data. While our framework provides a foundation for the development and testing of reconstruction algorithms, an efficient working system has yet to be developed.

The main contribution of this work is the design of a modular and extensible framework for vessel assembly. We validate this framework by demonstrating a prototype reconstruction algorithm that identifies likely pairs and triples of sherds. Our strategy utilizes a multi-feature match likelihood computed by independent evaluation modules and demonstrates how one can extend pair-wise match likelihoods to handle triples and larger configurations of sherds.

References

Cooper, D.; Willis, A.; Cao, Y.; Han, D.; Leymarie, F.; Orriols, X.; Mumford, D.; et al. 2001. Assembling virtual pots from 3D measurements of their fragments. In *VAST International Symposium on Virtual Reality Archaeology and Cultural Heritage*.

da Gama Leito, H. C., and Stolfi, J. 1998. Automatic reassembly of irregular fragments. In *Universidade Fedral Fluminense (Estadual de Campinas), Technical Report IC-98-06*.

Jepson, A., and Mann, R. 1999. Qualitative probabilities for image interpretation. In *ICCV (2)*, 1123–1130.

Keim, G.; Shazeer, N.; Littman, M.; Agarwal, S.; Cheves, C.; Fitzgerald, J.; Grosland, J.; Jiang, F.; Pollard, S.; and Weinmeister, K. 1999. PROVERB: The probabilistic cruciverbalist. In *AAAI / IAAI*, 710–717.

Keim, G.; Shazeer, N.; and Littman, M. 1999. Solving crossword puzzles as probabilistic constraint satisfaction. In *AAAI / IAAI*, 156–162.

NAG. 1993. *NAG Fortran Library*. 1400 Opus Place, Suite 200, Downers Grove, Illinois 60515: Numerical Algorithms Group.

Pankanti, S.; Jain, A. K.; and Tuceryan, M. 1994. On integration of vision modules. In *CVPR*, 316–322.

Papaioannou, G.; Karabassi, E. A.; and Theoharis, T. 2001. Virtual archaeologist: Assembling the past. In *IEEE Computer Graphics and Applications*, volume 21, 53–59.

Press, W. H.; Flannery, B. P.; Teukolsky, S. A.; and Vetterling, W. T. 1986. *Numerical Recipes: The Art of Scientific Computing*. Cambridge, UK: Cambridge University Press.

Ucoluk, G., and Toroslu, I. H. 1999. Automatic reconstruction of broken 3-d surface objects. *Computers & Graphics* 23(4):573–582.