

Towards Automatic Classification of 3-D Museum Artifacts using Ontological Concepts

Simon Goodall, Paul Lewis, and Kirk Martinez

Electronics and Computer Science,
University of Southampton,
SO17 1BJ,
United Kingdom
{sg02r,phl,km}@ecs.soton.ac.uk

Abstract. The development and use of content-based retrieval techniques for 3-D models is a relatively new departure in multimedia retrieval. We have extended our existing multimedia museum information system to support content-, metadata- and concept-based retrieval of 3-D models of museum artifacts and in this paper we describe a “classifier agent” to automatically assign associations between 3-D artifacts and concepts and metadata stored in a domain ontology. The context of the classifier agent is described, together with an overview of its architecture. Selecting appropriate parameters for the agent is an important activity and a comparison is made between manually selected parameters and the results of an automatic technique to determine “optimal” settings.

1 Introduction

An increasing number of museum systems are being developed to store and organise multimedia data on almost any subject. This has led to more and more sophisticated search and retrieval methods to access the data. However, adding new data to the system is still largely a manual task and in some cases classification information may not be complete. The European project, SCULPTEUR [1, 2], is concerned with the perceived benefit of structuring and integrating the knowledge associated with museum artifacts, enabling users to more fully exploit the richness of the data, facilitating more versatile browsing, retrieval and navigation within collections, and enabling cross collection searching and interoperability with external systems. Starting with the conceptual reference model (CRM) [3] developed by the museum documentation standards organisation, CIDOC, ontological descriptions of the museum collections have been developed. Metadata associated with the artifacts has been mapped to the ontology to form an integrated knowledge base. Graphical tools have been developed to provide browsing of the concepts, relationships and instances within the collections.

A novel aspect of the project is the ability to search and retrieve 3-D objects, in addition to 2-D image data, through a range of integrated methods. In addition to a standard textual search interface, objects can also be retrieved by browsing the ontology (*concept*-based retrieval) or by providing an example to

the system (*content*-based retrieval) or through a combination of these methods. For example a user can provide an example 3-D model of a vase and a text based entry of “Greek” to retrieve Greek vases that are of a similar 3-D shape to the example.

As part of its knowledge acquisition package, a classifier agent to automatically associate 3-D object models to concepts and metadata within the domain ontology is being developed. This paper focuses on the classifier agent, providing a description of its development and aspects of the parameter selection process.

The Search and Retrieval Web Service (SRW), a standard developed from the Z39.50 stable [4], provides the interface to these query mechanisms for both the internal SCULPTEUR components and external systems that understand the CIDOC CRM based ontology. The SRW is an important feature for integrating the classifier agent with the rest of the system.

The rest of this paper is organised as follows. The next section describes the classifier agent, its architecture and current status. Section 3 describes our approach to automatic parameter selection, section 4 and 5 provide details of experimentation and results respectively and section 6 provides pointers to related work. Finally in section 7 we provide conclusions and an outline of future work.

2 The Classifier Agent

Associating 3-D models and their 3-D shape feature vectors with appropriate class labels in the ontology provides a potential training set for automatic classification of unclassified objects. This is the role of the classifier agent in SCULPTEUR. Of course, for many classification tasks it is faster and more reliable for a curator to classify objects manually as they enter the system. Certainly this is true for more obvious labels such as vase or statue. However, use of a classifier may be able to help with understanding more subtle class differences in circumstances where less is known about the artifact under consideration; e.g. Greek versus Polynesian. Classifications also allow faster retrieval results to be returned to the user because feature vector similarities do not have to be re-computed. The classifications give indexed access to the objects.

The classifier agent’s functionality is currently directly available to the users so that, if they wish to explore whether certain classes of objects can be distinguished, or study the taxonomic properties of specific artifacts, they are able to do so by invoking the classifier and instructing it to train on particular classes and their feature vectors. For ease of use, techniques for automatically developing and training classifiers need to be employed.

2.1 3-D Descriptors

The content-based multimedia retrieval features of the system are facilitated through the use of feature vectors (or descriptors) extracted from the media objects. This is not only true for 2-D image data but also the 3-D objects now

being stored in the system. Several 3-D descriptor extraction algorithms have been implemented and integrated to provide 3-D content-based retrieval. These include the D2 shape distribution descriptors (Shape D2) from the Princeton Shape Retrieval and Analysis Group [5] and the histogram descriptors (Cord Hist 1, Cord Hist 2, Cord Hist 3, Cord Histogram) from Paquet and Rioux developed as part of the Nefertiti system [6]. An area to volume ratio descriptor (Area Volume) [7], which is a single valued statistic giving the ratio of the surface area of the model to its enclosed volume is also introduced to provide a fast discriminator which can reduce the search space. The Extended Gaussian Image (EGI) [8] and 3-D Hough Transform [9] have also been implemented. These descriptors have two versions based upon differing methods of partitioning the object space (EGI Oct, EGI Sphere, Hough Oct and Hough Sphere).

The 3-D descriptors not only support the 3-D content-based facilities of the system but also provide data for use by the classifier agent.

2.2 Distance metrics

In order to establish the similarity (closeness) of two feature vectors in some feature space, a wide range of distance metrics have been presented in the literature. The most commonly used are the Minkowski norms, typically the L_1 norm (the city block distance) and the L_2 norm (the Euclidean distance). (See e.g. [10, 5]). The norms are particularly attractive as they are simple to calculate and generally produce good results. However other distance metrics may provide better results when used in combination with specific descriptors and types of object.

Osada et al. [5] suggest a range of distance metrics that could be used for comparison purposes. These are the Kolmogorov-Smirnov distance, Kullback-Leibler divergence distance, Match distances, Earth Mover's distance and the Bhattacharyya distance. Hetzel et al. [11] suggest the histogram intersection and the χ^2 distance, while Ankerst et al. [10] suggest the Quadratic distance.

In the classifier agent several distance metrics have been implemented so that the best in terms of classifier performance may be selected. Those implemented include the city block, Euclidean, histogram intersection, Bhattacharyya, quadratic, Kullback-Leibler (both symmetric and non-symmetric) and the χ^2 distances.

2.3 Architecture

The classifier agent is organised around a collection of classifiers, some of which will be tailored to generally classifying between a large number of classes, and some which will be specialised to distinguish between a small number of classes. Over time, this will result in large numbers of classifiers tailored to specific datasets, and specialised in different areas. To use effectively, some prior knowledge is required by the user (or the system) in selecting the appropriate classifiers to obtain a correct classification: for example, if a user already knows that their

object is a vase, but is less sure about the specific type, they can use a classifier tailored to distinguishing between vase types rather than between broader classes. However this is not an issue addressed in this paper.

The system ontology contains a number of “classes” which indicate object type such as vase, statue, tile etc, but it also includes artists’ names, periods of creation etc. These different classes are not mutually exclusive, so that if the class labels are associated with distinctive descriptor sets, a query object may obtain several labels during the classification process. The agent is able to query the system ontology for these labels and retrieve URLs pointing to 3-D objects and feature vectors through the Search and Retrieval Web service (SRW) interface. The user can experiment with the agent to explore whether the descriptors do provide the classification capabilities required. The classifier agent can also use locally stored data. An XML file stores the class labels for an object, along with the location of its associated feature vectors.

2.4 Current status

Initially a large range of standard classification algorithms, distance metrics and adjustable parameters were explicitly available within the classifier agent. However, it became clear that for most users not versed in classification strategies, a limited set of options with automatic techniques for classifier development was the ideal.

The current version offers two very basic classification algorithms, a k -Nearest Neighbour (k -NN) method and a k -Means classifier [12, 13]. These techniques have been chosen as they are well known, easily understood and have the added advantage for the user that they function in a similar way to the content-based retrieval process, allowing users to gain a better understanding of how the different descriptors perform.

The feature vectors generated from various 3-D content-based retrieval techniques are used as the inputs to the classifiers (see section 2.1). Only single feature vector types are used as the input. However combining feature vectors in the classifier has the potential to improve performance.

The k -NN and k -Means classifiers both make use of a distance metric to compare different input patterns. Section 2.2 goes into further details about the distance metrics available in the agent.

Each classification scheme has parameters that can be set to adjust the classifier performance. The optimal settings are very dependant upon the data used to train the classifier and are typically not known in advance. For the k -NN classifier these parameters are feature vector type, distance metric and k . The k -Means classifier takes the same parameters in addition to a threshold used to decide when to terminate training. As an alternative to manually specifying parameters, an automatic parameter selection scheme is available. This facility is described in section 3.

The classifier agent is currently implemented using PHP to provide the user interface and C++ binaries perform the back end classification tasks. The interface allows users to manually train classifiers using an available dataset. Users

can also upload their own objects to the system to be classified by the classifiers that exist there. The agent provides an interface that allows a user to specify the parameters for which to train a classifier. Resulting classifications are displayed to the user along with a range of performance statistics.

3 Automatic Parameter Selection

An expert user of the system may wish to manually select the classification algorithm and its parameters, the distance metric and the 3-D shape descriptor type to be used. However, for most users, an automatic parameter selection scheme is preferred. To assist the user to use the classifier agent effectively, appropriate values need to be chosen automatically where possible. These depend on the dataset and to a lesser extent on the speed and quality required. In some cases prior knowledge can be used to estimate “good” parameter values, however more typically there will be little prior knowledge available.

Several techniques for automatically searching for optimal parameter values have been described in the literature. In the classifier agent we have implemented the classical particle swarm optimisation (PSO) algorithm [14] to search for appropriate parameters for the classification algorithms for a given training set.

PSO’s use a swarm of particles which represent points within parameter space. Each particle records its best position, and each particle has access to the global best position. At each iteration, the current performance of each particle’s parameters is recorded and the best position is updated if applicable. Each particle then updates its position based on how far away it is from both its personal best, and the global best, with the aim of moving closer to these positions. A random factor is introduced to avoid particles directly homing in on the centre point between the global and personal best.

Due to PSO’s searching through a continuous space, it cannot be used to find discrete values (such as the distance metric or feature vector type) and an exhaustive search is employed to set these variables before initiating a PSO based search of the remaining variables.

In order to reduce the search space, we can use properties specific to the classification schemes to limit the range in which a parameter can lie. For the k -NN classifier we can limit the upper size of k . Intuitively, we would expect that k should not be larger than the smallest class size in the training set, as larger values will become biased to the larger classes, and it can be seen in the results that the lower values of k perform better. By limiting k to the size of the smallest class, the search space is reduced significantly.

4 Experimentation

The classifier has been evaluated using a dataset composed of 144 manually classified museum objects. Table 1 provides information on the dataset.

As part of the prototype evaluation in SCULPTEUR, users were asked to evaluate the classifier agent. They were asked to create classifiers based on both

Table 1. The Museum Dataset

Class Name	Training	Testing
Statue	8	7
Vase	31	31
Tile	16	16
Misc	9	10
Mask	3	3
Tool	5	5

the k -NN and k -Means algorithms and attempt to use them to classify their own objects. This evaluation used the museum dataset. The resulting classifiers are presented here. As a comparison, classifiers created using automatic parameter selection are also presented for the museum dataset.

Previous work on analysing descriptor performance in [15] showed the the Area Volume descriptor gave the best performance results for the models obtained from the museums.

The *accuracy* statistic is used to evaluate classifier performance and is defined as $(TP + TN) / (TP + TN + FP + FN)$ where TP is the number of true positives, TN is the number of true negatives, FP is the number of false positives and FN is the number of false negatives. See [16] for more details on evaluating classifier performance.

Some early default values were defined for the classifier parameters based upon results from previous work [15] which showed that the Area Volume descriptor and Euclidean metric were good for this type of data.

For the particle swarm optimisation we used 10 particles and 10 iterations. During our experimentation, this ensured the swarm converged without too many unnecessary iterations.

5 Results

The classifier agent is still under development. However, some preliminary results have been obtained. These are results of classifiers created through manual parameter selection from the user evaluation, automatic parameter selection, and a summary of the results of the project evaluation which trialled the manual classification.

Table 2 shows the results for the k -NN classification scheme from both manual classifications obtained during the evaluation process, and from automatically generated classifiers obtained from several runs. Duplicate results have been omitted. The first line of the table shows the results of the default parameter values. The results show that the automatically generated classifiers have an increased accuracy of around 10% over the manually chosen ones and there is a large difference in the value of k chosen by the automatic parameter optimiser and the manual classifiers. The table also suggests that the choice of distance

Table 2. k-NN Results

Type	Descriptor	Metric	k	Accuracy
Manual	Area Volume	Euclidean	15	84.7%
Manual	Hough (Oct)	Euclidean	3	89.8%
Manual	Shape D2	Euclidean	15	87.9%
Manual	Cord Hist 1	Euclidean	15	70.3%
Automatic	Area Volume	City Block	1	97.6%
Automatic	Shape D2	City Block	1	98.1%
Automatic	Shape D2	Intersection	1	98.1%
Automatic	Cord Hist 1	Quadratic	1	96.8%

metric does not play a large role as the data in the case of the final two combinations for Shape D2 show the same accuracy, but with a different metric. The Shape D2 descriptor gave the best results, although they are only slightly better than the equivalent results for the Area Volume descriptor.

Previous work in [15] comparing 3-D shape descriptors and distance metrics on a similar manually classified museum dataset showed that the Area Volume descriptor performed best overall. However, the Shape D2 descriptor did best for the “nearest neighbour statistic” used in that analysis. This statistic indicates the proportion of all objects in the dataset for which the nearest neighbour in feature space is of the correct class. This is equivalent to the the k -NN classifier when k is equal to 1 and corresponds to some of the classifiers achieved by the automatic parameter selection technique.

Table 3. k-Means Results

Type	Descriptor	Metric	k	Threshold	Accuracy
Manual	Area Volume	Euclidean	15	0.1	87.0%
Manual	Area Volume	Intersection	15	0.1	82.8%
Manual	Area Volume	City Block	15	0.1	86.5%
Manual	Area Volume	Quadratic	3	0.1	82.4%
Manual	Area Volume	Euclidean	20	0.05	87.0%
Manual	Cord Histogram	Quadratic	15	0.1	71.7%
Manual	EGI (Sphere)	Euclidean	15	0.1	86.3%
Manual	Area Volume	City Block	15	1.0	87.0%
Manual	Area Volume	Euclidean	15	0.1	87.9%
Manual	Cord Histogram	City Block	20	1.0	83.7%
Manual	EGI (Sphere)	Chi	50	10.0	81.0%
Automatic	Area Volume	City Block	50	0.536	92.1%
Automatic	Area Volume	City Block	19	0.561	93.5%
Automatic	Shape D2	City Block	13	0.00	93.5%

Table 3 shows the results from the museum dataset for the k -Means classification scheme from both manual classifications obtained during the evaluation process, and from automatically generated classifiers obtained from several runs.

The first line of the table shows the results of the default parameter values. As with k -NN, the automatically determined parameters performed best. However, the difference in performance is not as great. Due to the nature of the k -Means algorithm, repeating the training process with the same parameters does not necessarily generate the same classifier, hence good values from the training set may not always produce good classifiers for the test set. This makes it harder to evaluate the effect of different parameters. The default parameters give reasonable results, but not the best. The automatically generated classifiers have reasonably consistent descriptor and distance metric values, however the value of k changes significantly.

The project evaluation presented the system to the user partners and asked them to both create and test classifiers and give feedback on the user interface. Typically the users found specifying parameters for the classifier confusing, either because they did not understand what the field was, or how the values would affect performance. This highlighted the need to do this automatically. A common complaint was that the presentation of several evaluation statistics was confusing and suggested a need for either a single statistic to represent the overall performance, or the use of some other more “friendly” indicator of performance or confidence.

6 Related Work

The work described has drawn on a substantial body of established work and more recent research performed by others. The 3-D feature vectors are a subset of recently published algorithms for 3-D shape representation and matching. We are particularly indebted to the Princeton work [17]. For a recent review of this area see Tangelder et al. [18].

There are many more advanced alternative approaches to the classification problem than the ones currently used in our classifier. See [12, 13] for an overview of classification techniques.

We chose to use the particle swarm optimisation algorithm for parameter optimisation but many alternative techniques could have been used including for example genetic algorithms [19] and simulated annealing [20].

In the work so far we have only considered individual classifiers. Combining classifiers is in our plan for future work and there has been much significant work in this area (e.g. [21]).

Several 3-D demonstrator systems have been built to be able to compare different 3-D descriptors [22–24]. However, it seems that few ontology centred multimedia retrieval systems for real applications have emerged with content based 3-D model retrieval as an integral part of the system.

The reader is referred to [1, 2] for further details of the various aspects of the SCULPTEUR project.

7 Conclusions & Future Work

The classifier agent in the SCULPTEUR system has been presented in terms of the overall design, current status and preliminary results. These show that automatic techniques for parameter setting result in better classifiers than those created manually by users. Our current approach can take a long time to determine the optimal parameters and investigating possibilities for more rapid strategies is an area for future work.

More recent work on PSO's has added the ability to search discrete parameters [25] and these will be investigated.

Other areas of future work include working with larger datasets, combining classifiers, investigating other classification techniques and introducing alternative 3-D descriptors. The automatic optimisation techniques may help in other areas of the agent such as determining appropriate weightings for combining classifiers.

Acknowledgements

The authors wish to thank: the European Commission for support through the SCULPTEUR project under grant IST-2001-35372. We would also like to thank our collaborators on the project, including Fabrizio Giorgini from Giunti Labs, Matthew Addis, Adrian Pillinger and Daniel Prideaux from IT-Innovation, Southampton, Francis Schmitt and Tony Tung of ENST, Paris, Christian Lahanier of C2RMF, James Stevenson and Rachel Coates of the V&A museum, Joseph Padfield of the National Gallery, Raffaella Rimaboschi of the Uffizi and Jean-Pierre of the Musée de Cherbourg for many useful discussions, use of data and valuable help and advice; and Hewlett Packard's Art & Science programme for the donation of server equipment.

References

1. Goodall, S., Lewis, P.H., Matrinez, K., Sinclair, P.A.S., Giorgini, F., Addis, M.J., Boniface, M.J., Lahanier, C., Stevenson, J.: SCULPTEUR: Multimedia Retrieval for Museums. In: Image and Video Retrieval: Third International Conference, CIVR 2004, Dublin, Ireland (2004) 638–646
2. Addis, M., Boniface, M., Goodall, S., Grimwood, P., Kim, S., Lewis, P., Martinez, K., Steveson, A.: SCULPTEUR: Towards a New Paradigm for Multimedia Museum Information Handling. In: International Semantic Web Conference (ISWC 2003), Florida, USA (2003) 582–596
3. Crofts, N., Dionissiadou, I., Doerr, M., Stiff, M.: Definition of the CIDOC Object-Orientated Conceptual Reference Model, v.3.1 (2001)
4. SRW Editorial Board: Zing search and retrieve web service. (<http://www.loc.gov/z3950/agency/zing/srw>)
5. Osada, R., Funkhouser, T., Chazelle, B., Dobkin, D.: Matching 3D Models with Shape Distributions. In: Shape Modeling International, Genova, Italy (2001) 154–166

6. Paquet, E., Rioux, M.: Nefertiti: a query by content system for three-dimensional model and image databases management. *Image and Vision Computing* **17** (1999) 157–166
7. Tung, T., Schmitt, F.: Augmented Reeb Graphs for Content-Based Retrieval of 3D Mesh Models. In: *International Conference on Shape Modeling and Applications 2004*, Genova, Italy (2004) 157–166
8. Horn, B.K.P.: Extended Gaussian Images. *Proceedings of the IEEE* **72** (1984) 1671–1686
9. Zaharia, T., Prêteux, F.: Hough transform-based 3D mesh retrieval. In: *Proceedings SPIE Conference 4476 on Vision Geometry X*, San Diego, CA (2001) 175–185
10. Ankerst, M., Kastenmüller, G., Kriegel, H.P., Seidl, T.: Nearest Neighbor classification in 3D protein databases. In: *7th International Conference on Intelligent Systems for Molecular Biology (ISMB'99)*, Heidelberg, Germany, AAAI Press (1999) 34–43
11. Hetzel, G., Leibe, B., Levi, P., Schiele, B.: 3D Object Recognition from Range Images using Local Feature Histograms. In: *IEEE International Conference on Computer Vision and Pattern Recognition (CVPR'01)*. Volume 2., Kauai Island, Hawaii (2001) 394–399
12. Bishop, C.M.: *Neural Networks for Pattern Recognition*. 4 edn. Oxford University Press (1997)
13. Haykin, S.: *Neural Networks: A Comprehensive Foundation*. Prentice Hall (1999)
14. Kennedy, K., Eberhart, R.C.: Particle swarm optimization. In: *Proceedings of IEEE International Conference on Neural Networks*, Piscataway, NJ (1995) 1942–1948
15. Goodall, S., Lewis, P., Martinez, K.: 3-D shape descriptors and distance metrics for content-based artefact retrieval. In Lienhart, R.W., Babaguchi, N., eds.: *Proceedings of Storage and Retrieval Methods and Applications for Multimedia 2005*, San Jose, California, USA (2005) 87–97
16. Kohavi, R., Provost, F.: Glossary of terms. *Special Issue of Applications of Machine Learning and the Knowledge Discovery Process* **30** (1998) 271–274
17. Shilane, P., Min, P., Kazhdan, M., Funkhouser, T.: The Princeton Shape Benchmark. In: *Shape Modeling International (SMI04)*, Genova, Italy (2004) 167–178
18. Tangelder, J.W.H., Veltkamp, R.C.: A Survey of Content Based 3D Shape Retrieval Methods. In: *International Conference on Shape Modeling and Applications 2004*, Genova, Italy (2004) 145–156
19. Beasley, D., Bull, D.R., Martin, R.R.: An overview of genetic algorithms: Part 1, fundamentals. *University Computing* **15** (1993) 58–69
20. Kirkpatrick, S., Gelatt, J., Vecchi, M.P.: Optimization by simulated annealing. *Science* **220** (1983) 671–680
21. Breiman, L.: Bagging predictors. *Machine Learning* **24** (1996) 123–140
22. Funkhouser, T., Min, P., Kazhdan, M., Chen, J., Halderman, A., Dobkin, D.: Princeton 3d model search engine. (<http://shape.cs.princeton.edu/search.html> [Accessed 2005-04-25])
23. Paquet, E., Rioux, M.: Nefertiti - content-based indexing and retrieval of 3-d and image databases. (<http://www.cleopatra.nrc.ca> [Accessed 2005-04-25])
24. Vranic, D.V.: Content-based classification of 3d-models by capturing spatial characteristics. (<http://merkur01.inf.uni-konstanz.cd/CCCC> [Accessed 2005-04-25])
25. Kennedy, J., Eberhart, R.C.: A discrete binary version of the particle swarm algorithm. In: *Proceedings of the 1997 Conference on Systems, Man and Cybernetics*, IEEE Service Center, Piscataway, NJ (1997) 4104–4109