The 10th International Symposium on Virtual Reality, Archaeology and Cultural Heritage VAST (2009)
K. Debattista, C. Perlingieri, D. Pitzalis, and S. Spina (Editors)

Reconstruction of large cultural heritage sites from archived maps

R. G. Laycock, S. D. Laycock, A. M. Day

School of Computing Sciences
University of East Anglia, Norwich, NR4 7TJ, UK
{rgl,sdl,amd}@cmp.uea.ac.uk

Abstract

Reconstructions of large cultural heritage sites, at multiple time periods, facilitate public awareness, enable the visualisation of regeneration proposals, and assist archaeologists in establishing the validity of a particular hypothesis within context. This paper focuses on the reconstruction of sites that no longer exist, where archived cartography and archaeologist's sketches provide an invaluable resource conveying the layout of an area. Whilst three-dimensional models are used in a broad range of applications their construction typically involves a labour intensive process and this paper presents a set of techniques to aid the reconstruction of environments from maps. In particular, the approach considers that an environment will exhibit a substantial amount of similarity, which is exploited to reduce the modelling time. The concept of similarity permits a dominant set of k building footprints to be identified from a map. A set of models representing the k dominant footprints are created and, based upon both the image based and geometry based metrics discussed in this paper, are aligned to the closest matching footprint in the archived map. Any building that is not sufficiently close to any of the k dominant footprints is labelled as being visually distinct and requires further modelling. To evaluate the technique a reconstruction of 19th Century Koblenz is undertaken, where 2300 building footprints are extracted, classified and aligned to one of 51 dominant building footprints in under fifteen minutes.

Categories and Subject Descriptors (according to ACM CCS): I.3.5 [Computer Graphics]: Computational Geometry and Object Modeling—Geometric algorithms, languages, and systems

1. Introduction

Recent advances in computer graphics hardware, and efficient techniques to exploit it, have facilitated real-time solutions to the traversal of large heritage sites. Visualizing a large cultural heritage site in its geographical and temporal context offers the possibility of virtual tourism without detrimental effect to the site and allows regeneration scenarios to be considered. Constructing three dimensional models of existing cultural heritage sites has received significant attention in two main categories: laser scanning, [STY*03], and photogrammetry, [KHN04]. Whilst these approaches have been used extensively to record, measure and preserve cultural heritage sites they are only capable of displaying the current state of the environment. This paper considers the reconstruction of sites that no longer exist in their entirety, which poses a significant challenge in terms of obtaining an accurate representation of the area.

A wide variety of resources are available for guiding the reconstruction process including oil-paintings, sketches, architectural drawings and maps. In the early to mid 19th Century European cartographers encoded building footprints into maps, which offers a potential source for deriving the layout of an environment. Utilizing effectively the limited and potentially conflicting resources describing the structure and appearance of a site is a challenging and often time consuming endeavour. Consequently, this paper considers techniques to aid in the reconstruction process, in particular, advancing towards the objective of rapidly modelling a 3D site from building footprints depicted in archived maps or archaeologist's sketches. These resources are frequently

DOI: 10.2312/VAST/VAST09/041-048

in raster format, but archaeological surveys may be vector based and therefore the techniques should consider both geometry based and image based solutions in order not to lose any of the limited information describing a site.

Obtaining a high fidelity reconstruction of a large cultural heritage site involves close collaboration between digital artists and historians, since all the available data must be analysed and used as a basis for the reconstruction. Achieving a high fidelity model is frequently the result of interactive modelling, as it provides the modeller with sufficient flexibility for creating the desired geometry and appearance within a reasonable time frame. However, interactively modelling a building for every extracted footprint is clearly prohibitively time consuming. Consequently, the similarity between buildings in an urban environment will be exploited to facilitate the rapid reconstruction of a three dimensional model, which can subsequently be refined in specific areas, as required by the application.

The main contribution of this paper is a set of techniques to identify a subset of building footprints that are most representative of the buildings in the environment. A building can subsequently be modelled for each of the dominant building footprints in the subset and automatically aligned to the footprint in the environment that matches with least error. Furthermore, an additional subset of important buildings with unique footprints will be determined automatically and presented to the user to facilitate the manual placement of key buildings or monuments. The automatic extraction and classification of building footprints from maps reduces the modelling time and via instancing improves rendering performance for both offline and real-time visualisations.

2. Previous work

Utilising archived maps for reconstructing eradicated or partially destroyed environments has received limited attention from computer scientists, despite the wealth of archived resources that exist. One of the fundamental issues that remains is to reduce the manual processing time required to extract information describing the layout of the environment. Frequently archived resources are subject to noise and contain many overlapping objects representing different cartographic elements. This hinders the task of automatic or computer assisted extraction, which has largely been circumvented in the literature.

Procedural modelling techniques have been developed to create large environments based on building layouts. Muller et. al, [MWH*06] reconstructed the city of Pompeii using a structured grammar. However, only a small section was reliably aligned to the manually defined building footprints. The remaining buildings were procedurally generated amongst the street network based on techniques presented in [PM01]. An approach is required to automatically extract the building footprints to facilitate the technique's application to large scale environments. Recently, CityEngine was applied to increase the fidelity of the laser scanned plaster of paris model for Rome Reborn, [GFDS*05]. However, in this work, the building footprints were created by converting the initial scanned 3d model into a coarse representation, which could be input into the CityEngine system.

Birch et al., [BBJ*01], developed an interactive modelling system capable of generating general built environments. Their modelling approach coupled a shell modeller with a window modeller to enable users to rapidly construct buildings, of a variety of architectural styles, comprising a number of storeys and facade features. The technique was used to reconstruct a medieval town, where building layouts were not available and therefore the buildings were subsequently positioned by aligning them to a Bézier curve, manually defined to follow the streets on a scanned historic map. This approach facilitated the rapid definition of the environment's layout, but it did not consider accurate placement of buildings.

Suzuki and Chikatsu, [SC03], investigated the automatic extraction of building plots from archived maps depicting Kawagoe, Japan in the Edo period. Their approach utilised the map to obtain the position of rectangular footprints by automatically extracting the four corners. This approach is limited in that it only recovered rectangular footprints. Removing this limitation, Laycock et. al, [LDD08], considered recovering building footprints represented by non-intersecting closed polygons. However, this introduces further complications, since a polygon extracted from an image may have unit length edges. These require refinement such as the removal of near collinear points or the complete replacement with the largest oriented empty rectangle that can fit within the polygon.

Whilst fitting rectangular footprints provides a robust solution in the presence of noise and occlusion due to other cartographic elements, it has the limitation in reliably representing a large set of buildings. In the proposed work a dominant set of building footprints and a set of unique footprints are automatically identified. This allows for the footprints to be cleaned based on domain specific knowledge of the site, without requesting it a prior from the user. Furthermore, an interactive modelling package may be used to reconstruct a set of template models that can be automatically aligned to the building footprints in the scene in the presence of noise.

3. System overview

The techniques presented in the following sections are initiated on an archived two dimensional map representing the layout of an environment. This map may result from digitised maps, scanned paper-based sketches or archaeologist's diagrams constructed via an interactive editor. Therefore, the following techniques are derived for both image based and geometry based inputs. The approach commences by identifying a set of feature vectors, representative of a building footprint, for all footprints depicted in the input. In the image case the feature vector constitutes a selection of pixels representing a buildings and in the geometry case it is comprised of a set of vertices ordered anticlockwise to form a polygon. The process of segmenting the relevant information from archived maps is application dependent and its description is discussed in Section 7 for the case study of reconstructing 19th Century Koblenz, Germany.

In the image based scenario the resulting groups of pixels may contain holes due to obscurity by text or other cartographic symbols. Furthermore, the boundary may not reliably represent the edges of the footprint, owing to the limited pixel resolution. Therefore, a technique is required to identify the footprints used for the buildings in the presence of these cases of potential error. Section 4 details how a dominant set of template building footprints can be extracted from the set of all image based and geometry based footprints.

The template building footprints are exported in AutoCAD, .dxf, format to facilitate their import into AutoCAD or a modelling package such as 3DS Max. Once in the chosen interactive modelling package, a model for each template footprint can be created. Section 5 describes how the template footprints can be aligned to the closest building footprint in the environment, whilst minimising the error between them. For 3DS Max, a MaxScript is created that reads the parameter information for rotation, translation and scale, to allow instances of the template model to automatically populate the scene. In addition, a set of visually distinct footprints are exported by the program, since these will be used to guide the interactive construction of the landmark buildings or monuments.

4. Footprint classification

Footprint classification proceeds from a set of feature vectors representing the building footprints with the objective of extracting a dominant set of feature vectors plus an additional set of distinct feature vectors. The classification is achieved using a variant of Lloyd's iterative refinement heuristic for k-medoids clustering. A medoid is defined as the feature vector in the cluster which is closest to the mean feature vector. One of the k-medoids defines the most central feature vector in a cluster and therefore, by definition, acts as a good candidate for a template building footprint. To facilitate the modelling of each template building footprint, it is desired that a minimal set be selected. Therefore, the user specifies the maximum value permitted for k and the algorithm selects the number of clusters that causes the minimal root mean square error between each template footprint and the footprints in the corresponding cluster. The constraint that the number of clusters, k, selected causes minimal error is relaxed within a user specified tolerance, to permit smaller cluster sizes to be favoured.

For each value of k, the algorithm initialises k clusters, each containing a single feature vector. The initialisation is undertaken using a heuristic to improve the quality of the start state. The first medoid candidate is selected as the feature vector which is the most dissimilar to all others in the set. The i^{th} medoid candidate is chosen as the farthest from the previously chosen candidates. From this starting set of k-medoids, the remaining feature vectors are classified by assigning them to the set which has the most similar medoid. Similarity is calculated based on a distance metric, which depends on the content of the feature vector and specific details are provided in subsection 4.1 for geometry based footprints and in subsection 4.2 for the image based case. The result provides an initial solution to the classification, where a set of k representative feature vectors have been determined, in addition to assigning each feature vector to the nearest representative feature vector. The total error in the solution is subsequently calculated by summing all the distances between each feature vector and its nearest medoid.

The initial solution is refined, to reduce the total error, through iterative selection of a randomly selected non-medoid feature vector, f'. By replacing one of the previously selected medoids with f' a new value for the total error can be calculated. Prior to the calculation all the feature vectors are reassigned to their nearest medoid, which may have altered due to the addition of the new medoid f'. If the act of using the new medoid causes a reduction in the total error then this solution becomes the current solution and the process continues to iterate until there is no change in the medoid selection. The final clustering of footprints is input to the footprint template matching system, described in Section 5, to identify the rigid body transformation required to align each footprint with the nearest one from the dominant set of template footprints. The following two subsections provide details on the distance metrics required to define the notion of similarity between geometry based and image based feature vectors.

4.1. Geometry based metric

The objective is to determine the similarity between two feature vectors, v_1 and v_2, each composed of a set of n two-dimensional points representing a polygonal footprint in the form $[(x_1, y_1)\dots(x_n, y_n)]^T$. This is achieved using Procrustes analysis, [DM98], since it facilitates the definition of similarity between two point sets by first aligning them into a common coordinate frame. Alignment of the two point sets begins by calculating the centroid of each point set to enable them to be translated to the origin using tX, tY, as defined in Equation 1. After the translation has been removed Equation 2 states the scale factor required to normalise each point set using the Frobenius norm.

$$(tX, tY) = \left(-\frac{1}{n}\sum_{j=1}^{n} x_j, -\frac{1}{n}\sum_{j=1}^{n} y_j \right) \qquad (1)$$

$$(s) = 1.0/\sqrt{\sum_{j=1}^{n}\left[\left(x_j+tX\right)^2+\left(y_j+tY\right)^2\right]} \qquad (2)$$

The final stage removes the rotational differences between the two point sets. The translated and scaled point sets form two $n \times 2$ matrices, $v_{1'}$ and $v_{2'}$, where the first column represents the x coordinates and the second column contains the y coordinates. The rotation matrix required to align $v_{1'}$ onto $v_{2'}$ is extracted by performing Singular Value Decomposition on the matrix $v_{1'}^T v_{2'} = UDV^T$. The 2×2 rotation matrix VU^T is multiplied by each point in $v_{1'}$ to rotate it into alignment. Solving for the 2×2 rotation matrix using SVD permits the matrix to contain appropriate scaling values should the geometry require to be reflected before rotating it into alignment. The distance can subsequently be calculated by determining the root mean square error between each of the vertices in one feature vector with their corresponding vertices in the other.

In this application footprints are only compared if they consist of an identical number of vertices; an infinite distance is assumed to exist between two footprints which differ in their number of vertices. A preprocessing stage to the algorithm removes collinear and collocated vertices to ensure that vertices are located on corners of the footprints. Furthermore, curved sections of the building footprints are resampled by recursively inserting vertices between two corner vertices such that the distance between the vertex and the line segment between its two neighbours is below a user specified value. Therefore, the optimum solution can be obtained for a given pair of geometry based footprints by rotating the vertex indices of one footprint until the minimum distance between the two is determined.

4.2. Image based metric

The image based metric is required to determine the distance between two feature vectors, x_1 and x_2, each comprised of a set of n pixels representing the building footprints, in the form $[(x_1,y_1)\ldots(x_n,y_n)]^T$. Applying Procrustes anaylsis to two sets of points poses a computational problem, since the correspondences between pixels is not known and enumerating all possibilities between two sets is infeasible. It is possible that a set of features could be extracted on the boundary of the pixel regions but these are prone to small fluctuations due to overlapping cartographic elements and the finite pixel resolution. Consequently, a distance metric based upon central moments is considered.

Hu [Hu62] proposed a set of seven moment invariants, which are invariant to translation, scale and rotation. The moment invariants are based upon scale-normalised centralised moments of up to order 3, as defined by Equation 3. μ_{pq} states how the centralised moments are computed based on the centroid of the feature points, $(\bar{x},\bar{y})$.

$$\begin{aligned}\mu_{pq} &= \sum_{j=1}^{n}\left(x_j-\bar{x}\right)^p\left(y_j-\bar{y}\right)^q \\ \eta_{pq} &= \frac{\mu_{pq}}{\mu_{00}^{\left(1+\frac{p+q}{2}\right)}}, \text{where } (p+q)\geq 2 \end{aligned} \qquad (3)$$

From the seven moment invariants the first six are used, expressed in Equations 5 to 9, since the seventh invariant describes skew invariance and hence this would distinguish a difference between two reflected images of an identical footprint. When the footprints are input into the system, the invariants are calculated and the difference between two footprints is obtained during clustering by calculating the Euclidean distance between two sets of moment invariants.

$$\begin{aligned}
I_1 &= \eta_{20}+\eta_{02} && (4)\\
I_2 &= (\eta_{20}-\eta_{02})^2+4\eta_{11}^2 && (5)\\
I_3 &= (\eta_{30}-3\eta_{12})^2+(3\eta_{21}-\eta_{03})^2 && (6)\\
I_4 &= (\eta_{30}+\eta_{12})^2+(\eta_{21}+\eta_{03})^2 && (7)\\
I_5 &= (\eta_{30}-3\eta_{12})(\eta_{30}+\eta_{12}) \\
&\quad \left[(\eta_{30}+\eta_{12})^2-3(\eta_{21}+\eta_{03})^2\right]+ \\
&\quad (3\eta_{21}-\eta_{03})(\eta_{21}+\eta_{03}) \\
&\quad \left[3(\eta_{30}+\eta_{12})^2-(\eta_{21}+\eta_{03})^2\right] && (8)\\
I_6 &= (\eta_{20}-\eta_{02})\left[(\eta_{30}+\eta_{12})^2-(\eta_{21}+\eta_{03})^2\right]+ \\
&\quad 4\eta_{11}(\eta_{30}+\eta_{12})(\eta_{21}+\eta_{03}) && (9)
\end{aligned}$$

4.3. Scale based clustering

The geometry and image based metrics are both invariant to scale and therefore clusters will be formed containing building footprints of the same shape at different sizes. The template model should be representative of the entire set of footprints within its cluster without performing any uniform scaling. Consequently each cluster is further subdivided into multiple clusters, such that each new cluster represents the same shape and differs by a user specified threshold in scale. The threshold of 30% of the footprint size was chosen for the tests presented in this paper and it can be chosen to tradeoff between reducing the set of template building models and the error in approximating a footprint with a template.

Scale based clustering is performed by first sorting all footprints in a given cluster based on their size, followed by two linear scans of the ordered footprints starting from the medoid and directed in opposing directions. Each scan moves through the ordered footprints until a footprint is encountered that is larger than the medoid by a user specified threshold. At this point a new medoid is created and the values of the footprint indices that are within the threshold of this footprint's size are removed from their original cluster and inserted into the new cluster. The process continues until all footprints have been reclassified based on scale.

5. Footprint template matching

The footprint classification algorithm identifies a set of template building footprints that represent a large proportion of the buildings in the site. These template building footprints are output in AutoCAD .dxf format to enable a building to be constructed for each template. This allows the digital artist to spend time modelling a limited set of buildings appropriate for each template footprint. In order to populate the environment with these buildings a set of parameters describing the rigid body transformation necessary to align the template to the building footprints from the map are required. The classification from Section 4 clustered the building footprints into sets with the common property that all building footprints in a set are closest to a single template footprint than any of the other templates. Therefore, it remains to calculate the alignment for a footprint in the set onto the associated template footprint with minimal error.

Subsection 5.1 provides details for how this was achieved for the geometry and image based feature vectors. The parameters for translation, scale and rotation along with an identifier for the template footprint are exported for each building footprint, as a separate record. This file is read by MaxScript code to automatically create an instance of the appropriate model referred to by the identifier and transform it into the appropriate position. For each record an instance is created and the transformation is applied to the object. Once the instance is created, (to maintain a small file size, an xref can be used instead), it is translated to the position indicated in the record. The rotation is applied around the z-axis and any scales are applied necessary to cause a reflection of the geometry. The following subsection details how these parameters are retrieved depending on geometry- or image-based feature vectors.

5.1. Feature vector alignment

The translation is recovered directly from the distance metric. The geometry based translation parameters are, $(-tX,-tY)$, which position the object at the correct location in the environment. The translational parameters for the image case are directly retrieved from the scale-normalised centralised moments, $(tX_I = \mu_{10}/\mu_{00}, tY_I = \mu_{01}/\mu_{00})$. Rotation may be derived from both the rotation matrix computed in the geometric distance metric and from the second moments in the image based metric. However, when the second order moments become close to zero the accuracy of the computed angle of rotation becomes uncertain due to noise. Therefore, Principal component analysis, PCA, is performed on the two feature vectors to be aligned.

The principal components are extracted by computing the eigenvectors from the covariance matrix of the feature vector. Once the two dominant axes for each feature vector are known, a rotation required to align one axis onto the other can be computed. For rotationally symmetric objects the dominant axes have the same gradient but maybe directed in the positive or negative direction. Furthermore, the dominant axes of an object, which is a reflected version of another object, can not be rotated into alignment. To counteract both of these issues one object's feature vector is scaled and rotated onto the other and the root mean square error between values is determined. The values for orientation and scale that provide the least error are recorded and used for footprint alignment in 3DS Max.

6. Results for footprint classification and alignment

To test the geometry based classification and alignment, a set of non-intersecting closed polygons were constructed using varying numbers of vertices. Instances of these polygons were reflected, rotated and translated to spread them out across a scene. Figure 1 illustrates the extraction of the eight footprint types, colouring each footprint based on its inclusion in a specific set. Eight template footprints are extracted and a set of models were created by extruding the template footprints for test purposes. Each extruded model is automatically aligned to the appropriate footprint.

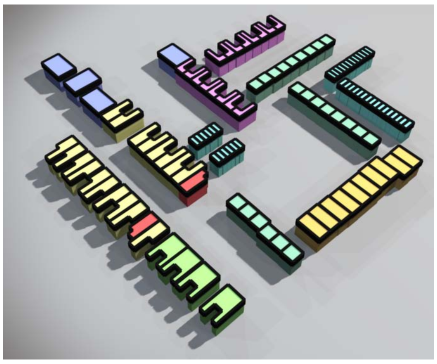

Figure 1: *A test scene comprising of geometry based footprints is constructed from reflected, rotated and translated polygons. A common set of eight template footprints are extracted, extruded and aligned automatically to the footprints in the scene.*

The process of image based classification and alignment is illustrated on a test image consisting of rotated, translated and reflected objects on a white background. To facilitate the segmentation all the objects were disjoint, to enable them to be extracted as a single connected component. Figure 2 illustrates the identification of the six unique objects and shows how a 3DS Max modelled object corresponding to each is aligned to the appropriate footprint automatically.

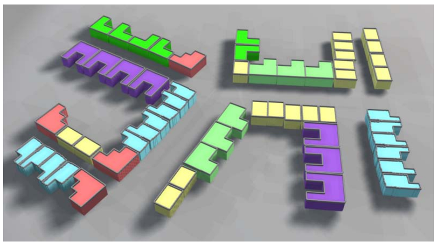

Figure 2: *A test scene comprising of image based footprints is constructed from rotated, translated and reflected collections of pixels. A common set of six template pixel groups are extracted and the corresponding model is aligned to the footprints in the scene.*

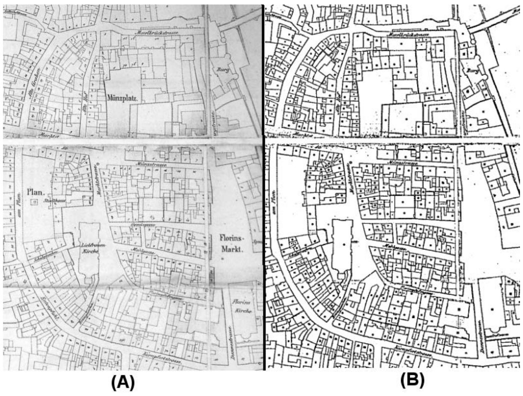

Figure 3: *(A) illustrates a photograph of part of an archived map representing the German city of Koblenz in the 19th Century. (B) presents the result after automatically pre-processing the image to obtain a binary image.*

7. Case study: reconstructing 19th century Koblenz

To illustrate the utility of the aforementioned techniques, a real world example of reconstructing 19th Century Koblenz, Germany is presented. The layout of the environment is depicted in a digitised map, as presented in Figure 3(A), where the locations of buildings are defined with black outlines. It illustrates the existence of noise in the image which is required to be cleaned. To achieve this the difference between multiple Gaussian filters applied across the image is taken followed by a remapping of the grey levels. Therefore enabling a binary threshold to segment the image into black boundary pixels and white background pixels. Figure 3(B) presents the results of applying this first pre-processing step to the grey level image to obtain a binary image.

To facilitate the identification of the building footprints the user is required to indicate any building footprints for extraction by placing a dot inside them. This is quickly achieved and these dots are automatically identified by performing a connected component analysis on the image. Any connected region of black pixels that is sufficiently circular is chosen as a building indicator point, the connected pixels are converted to white and their centroid is inserted into a list for further processing. The connected component analysis also enables the removal of regions of black pixels which are considered to be too small to be part of a building footprint.

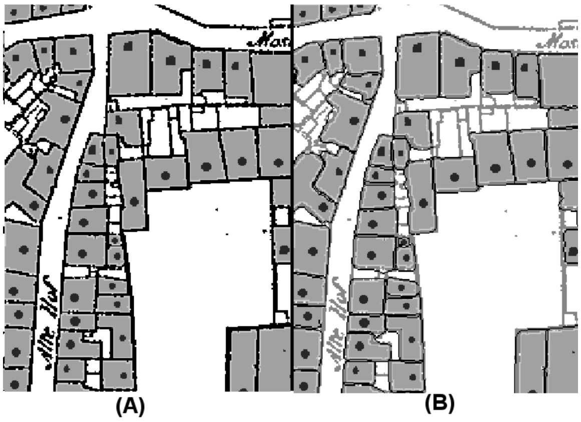

Figure 4: *(A) From each black building indicator point a seed fill algorithm is performed to construct the dark grey building areas. The initial building footprint polygons are shown in light grey surrounding each building area. (B) By employing an active contour model the footprints are adjusted to fit to the centres of the building footprint edges in the map.*

From each of the building indicator points a seed fill algorithm is initialised, which will terminate on reaching a black pixel in the binary image. Consequently, to ensure the seed fill algorithm does not escape through any small holes in the footprints, which may arise due to noise, a dilation is performed. A polygon for each seed fill region is defined by performing a boundary walk of the connected set of pixels. Figure 4(A) illustrates the regions in dark grey which have resulted from filling from the black building indicator points. The initial building footprint polygons are shown using light grey, partitioning the dark grey building footprint regions from the black building footprint pixels.

The dilation stage of the algorithm has increased the thickness of the footprint pixels resulting in reducing the area bounded by the building footprints. Consequently, a refinement stage is undertaken employing an active contour model to expand each of the initial building footprints away from its centroid, without allowing it to penetrate any other footprint. This is achieved by applying a distance transform to the binary image that indicates a high value for black pixels which are farthest from the background white pixels. Therefore the active contour model is able to expand the initial building footprint polygon while the movement causes the vertices to move to a pixel value with a higher distance value; resulting in the polygon finishing on the centre of the edges in the binary image as indicated in Figure 4(B).

Each building footprint is scan converted into an image to create a set of pixels, representing a feature vector. The image based metrics discussed previously are used to classify the footprints into sets and Figure 5 illustrates the result of the classification. Any building footprints that are sufficiently dissimilar to any of the template footprints identified are collected into a visually distinct set. The distinct footprints are shown in Figure 5 using a black outline. For this case study, they constitute two of the churches in Koblenz, which will require additional 3DS Max modelling.

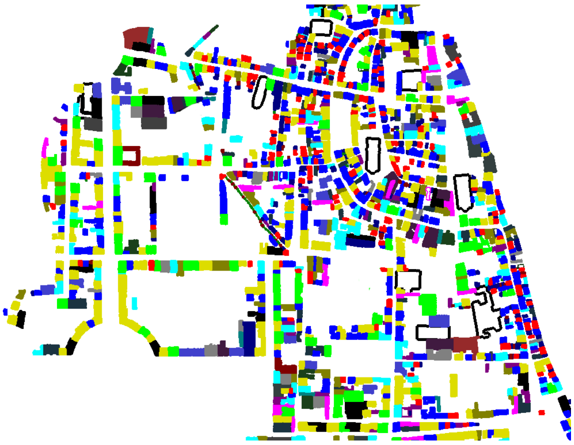

Figure 5: *The classification of the building footprints into 51 sets is illustrated by colouring all pixels included in a specific set with an identical colour.*

A set of landmark buildings were modelled for the visually distinct set of footprints in conjunction with a set of building models to represent the template footprints. Once modelled, the MaxScript was executed to populate the site with instances of the template buildings. Figure 6 illustrates a render from 3DS Max of the reconstruction.

8. Real-time rendering

While offline rendering permits high-fidelity reconstructions to be visualised, certainly the level of interactivity will increase a user's sense of immersion, [LCC*07]. An improved sense of immersion is important to a variety of applications, where it aids in increasing a virtual visitors understanding of an environment. Therefore, a real-time rendering component is implemented to efficiently render a large cultural heritage site using state of the art graphics techniques. During the reconstruction of Koblenz, Germany, 51 buildings were modelled and used to populate the 2300 extracted building footprints. In 3DS Max these buildings were instanced throughout the environment using MaxScript. In an analogous approach these buildings are translated and rotated into the correct location within an OpenGL framework in real-time.

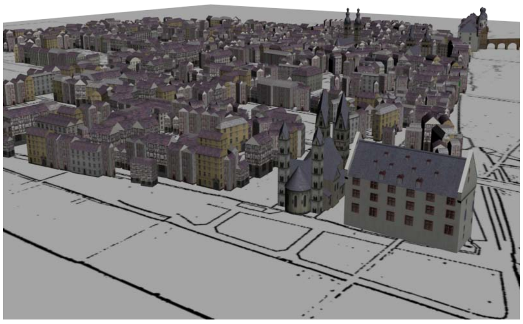

Figure 6: *Render of reconstructed 19th Century Koblenz, Germany*

The real-time rendering is achieved using geometry instancing, where multiple instances of an object are rendered after being transformed by a vertex shader. In this application geometry instancing is achieved by storing all the instances' positions and rotations in an array of floating point values. This array is created in a linear pass over all instances such that instances of a particular object are in a contiguous block in the array. In addition, another array of integers is maintained to store the size of these contiguous blocks. During rendering the floating point array is bound to the vertex shader along with an integer variable specifying the particular object being rendered. A call to $glDrawElementInstancedEXT$ with the desired number of instances permits the correct number of instances of the currently bound vertex buffer object to be created. In the vertex shader these instances are positioned correctly by retrieving the desired transformation from the floating point array. This is achieved by indexing based upon *gl_InstanceID*, the automatically generated instance number, offset by the number of instances that have currently been rendered. The approach permits the scene to be visualised at an average 260 frames per second.

9. Results

A Pentium 2.4GHz PC with 3GB of RAM was used for all tests in this work. The reconstruction of Koblenz required approximately 15 minutes to extract 2300 building footprints. In less than one minute 51 dominant template footprints were extracted. These footprints were subsequently imported into 3DS Max and individual building models were created for each. Further work was required to model the visually distinct buildings, automatically identified by the system. The two churches were placed on the footprints and additional models for the bridge and gate houses were introduced from archived images. By providing a unique identifying name to each object, the MaxScript automatically positions the objects throughout the scene in alignment with the footprint locations. To increase the diversity amongst the models in the scene, multiple versions for each dominant footprint could be created and randomly selected by the MaxScript.

Alternatively, a library of textured building facades, of various heights and aspect ratios, could be created in 3DS Max and used to automatically cover the perimeter of the template models. The facade geometry would be unlikely to tile exactly along each polygon edge of the footprint and therefore a plane with a suitable default wall texture is created to fill any small gaps. This technique permits different template models to be created automatically from a set of facades and eases modelling efforts as the number of template models increases. Furthermore, this approach lends itself well to real-time rendering, since the facade geometry could be reused via instancing, as opposed to instancing the entire building. This permits more effective exploitation of the similarity that is exhibited both globally across the environment and locally on a per building facade basis.

10. Conclusions and future work

This paper presents a set of techniques which advance towards the segmentation of building footprints from archived maps and archaeologist's sketches. In particular, data mining is performed using a variant of Lloyd's iterative refinement heuristic for *k*-medoids clustering to extract a dominant set of building footprints and a set of visually distinct footprints. These footprints are output to the user to permit a model to be constructed for each. The interactive modelling of the template models or building facades is performed by digital artists in conjunction with historians and the resultant models are subsequently automatically aligned onto the footprints in the map. This process reduces the modelling time required for an entire site facilitating the reconstruction of a high-fidelity environment.

Whilst this process reduces the modelling time, it is also beneficial in terms of reducing the memory overhead and offering optimisation for real-time rendering applications. The techniques are illustrated on a set of test cases as well as for an area of 19th Century Koblenz, Germany. The latter was processed automatically, after the initial interactive placement of building identification points, in under fifteen minutes requiring additional time for modelling in 3DS Max. It is the focus of future work to extend the segmentation algorithms to be applicable to more varied map types, with the aim to reduce manual intervention. Furthermore, extensions to the algorithms will be investigated to increase the utility of the approach. For example, many footprints may be slight variations of their closest matching template footprint and therefore the ability to morph a footprint into alignment will be considered. Maxscript could be employed to adjust the corresponding template model or facades could be tiled along the edges of the adjusted footprint. The facade tiling technique offers further opportunities to increase scene diversity by altering a building's appearance in the vertical axes, as well.

Computer-assisted modelling of cultural heritage sites permits large geographically and temporally located environments to be created. These are suited to a wide range of visualisation applications including virtual tourism, regeneration proposals and education. Furthermore, by reconstructing multiple time periods possible reconstruction hypotheses for a landmark building can be visualised in context, in addition to, aiding the understanding of urban planning principles.

11. Acknowledgements

The research leading to these results has received funding from the European Community's Seventh Framework Programme (FP7/2007-2013) under grant agreement no 231809.

References

[BBJ*01] BIRCH P., BROWNE S., JENNINGS V., DAY A., ARNOLD D.: Rapid procedural-modelling of architectural structures. In *VAST 2001* (Athens, November 2001), ACM, pp. 187–196.

[DM98] DRYDEN I., MARDIA K. V.: *Statistical Shape Anaylsis*. John Wiley, Chichester, 1998.

[GFDS*05] GUIDI G., FRISCHER B., DE SIMONE M., CIOCI A., SPINETTI A., CAROSSO L., MICOLI L., RUSSO M., GRASSO T.: Virtualizing ancient rome: 3d acquisition and modeling of a large plaster-of-paris model of imperial rome. *Videometrics VIII 5665* (2005), 119–133.

[Hu62] HU M.: Visual pattern recognition by moment invariants. *IEEE Transactions on Information Theory 8* (1962), 179–187.

[KHN04] KIM Z., HUERTAS A., NEVATIA R.: Automatic description of complex buildings with multiple images. In *Computer Vision and Image Understanding* (2004), vol. 96, pp. 60–95.

[LCC*07] LAING R., CONNIFF A., CRAIG T., DIAZ C., SCOTT S.: Design and use of a virtual heritage model to enable a comparison of active navigation of buildings and spaces with passive observation. *Automation in Construction 16(6)* (2007), 830–841.

[LDD08] LAYCOCK R., DRINKWATER D., DAY A.: Exploring cultural heritage sites through space and time. *ACM Journal of Computing and Cultural Heritage 1(2)* (2008).

[MWH*06] MUELLER P., WONKA P., HAEGLER S., ULMER A., GOOL L. V.: Procedural modeling of buildings. *In Proceedings of ACM SIGGRAPH 2006 / ACM Transactions on Graphics (TOG) 25*, 3 (2006), 614–623.

[PM01] PARISH Y., MUELLER P.: Procedural modeling of cities. ACM Press, pp. 301–308.

[SC03] SUZUKI S., CHIKATSU H.: Recreating the past city model of historical town kawagoe from antique map. *Archives of Photogrammetry and Remote Sensing XXXIV-5/W10* (2003).

[STY*03] STUMPFEL J., TCHOU C., YUN N., MARTINEZ P., HAWKINS T., JONES A., EMERSON B., DEBEVEC P.: Digital reunification of the parthenon and its sculptures. In *VAST 2003* (Brighton, UK, 2003), pp. 41–50.