ETH
Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Institute of Geodesy and Photogrammetry igp

SWISS FEDERAL INSTITUTE OF TECHNOLOGY
Institute of Geodesy and Photogrammetry
ETH-Hoenggerberg, Zürich

RE-SEQUENCING A HISTORICAL PALM LEAF MANUSCRIPT

Prepared by:
M. Devrim AKCA

May, 2003

TABLE OF CONTENTS

1. INTRODUCTION

66 Historical Indian palm leaves, which were produced in the 8th Century AD, are kept in the *Museum Rietberg, Zürich*. On the leaves, there are figures and a long poem, inscribed in ancient Sanskrit language and narrating a love story. The original sequence of the leaves was lost long time ago. At one point in history, the stack of leaves was damaged by a mouse. Only the first 18 leaves have their pages numbered in Sanskrit language, but the rest of them got out of order. If it is assumed that the mouse ate the leaves in a regular manner, the geometry of the leaf perimeter, as left over after eating, should bear useful information to find the original sequence.

After acquiring digital images of the leaves and a pre-processing phase, an *inner boundary-tracing* algorithm was applied to all leaves in order to segment them. The fundamental data used in this work are boundary coordinates of the leaves. In order to obtain quantitative shape similarity measures, two different Boundary Based Shape Descriptor algorithms were applied to the boundary data: *Fourier descriptors* and a rotation-translation invariant *boundary intersection-based shape descriptor*. Shape descriptors indicate the similarity of different leaves. These similarity measures among all of the leaf pairs were arranged in the form of a symmetric square matrix. With this matrix and a threshold similarity value one can determine the most probable ancestor and successor leaves for a pointed leaf.

In the final step, a *Tree Search* scheme that starts from the 18th (fixed) leaf and ends at the 66th (relaxed) leaf was established to generate the most probable sequence. Every node in the tree was defined as a leaf and branched to the most probable neighbour leaves. The similarity measures were expressed as costs of the arcs, which connect two nodes in the tree. The sequence which has minimum total path cost was proposed as the most probable original sequence.

In Chapter 2, a brief information about the history of the leaf manuscript is given. In Chapter 3, the adopted methodology used to predict the most probable sequence is described. The obtained results are interpreted in Chapter 4. Finally, Appendix part gives the texture-mapped model of the leaf manuscript to provide visual inspection of the proposed sequence as well as the full list of the sequenced leaf images.

2. PROBLEM DEFINITION

In the collection of Museum Rietberg Zürich is an old palm leave manuscript from India, consisting of 66 folios, inscribed on both sides. Originally, they have all the same dimensions and were once tightly bundled. On them are 100 erotic poems by a 8th century AD Sanskrit poet (by the name of Amaru) inscribed, many of which are illustrated with one or more pictures. The manuscript was prepared about 200 years ago to be kept as a bundle. All folios have a hole in the

center through which a string was drawn to tie all leaves tightly between two wooden boards. However, the sequence of these poems is unknown (Fischer, 2002).

The manuscript was kept for two centuries in an Indian library. Some time ago, a mouse did start to eat part of these folios. It is likely that at that time, the right sequence was still established, which was later lost, except at the beginning: Folios 1-18 are paginated and therefore their sequence is known.

The mouse did not destroy the entire manuscript. She nibbled only about 5-10 % away from each leaf, mostly at the left side (Figure 1). If it is assumed that the mouse ate the leaves in a regular manner, the geometry of the leaf perimeter, as left over after eating, should bear useful information to find the original sequence.

Since the rest of the leaves do not give any information about the sequence, only the geometry of the harmed part is focused in order to avoid irrelevant data. The holes in the center of the leaves constitute a base point that is approximately at the same position on the all leaves. After the pre-processing phase i.e. rectification, the harmed left parts were cropped from the full images, and saved as different image files. All processes were performed based on these cropped images.

Two main process steps were adopted to achieve the solution. In the first phase, quantitative similarity measures, which will be addressed in the next chapter, were calculated among the all leaves. In the second phase, shape similarity measures are evaluated in a Tree-Search scheme to find the most probable sequence.

Figure 1: Left sides of the leaves were eaten by the mouse (front face).

3. METHOD

After acquiring the digital images of the leaves, some of the shape descriptor algorithms were applied to find the similar leaves. In the below, basic process steps are given.

- Taking the digital images of the leaves.
- Rectifying the images to correct the perspective effect due to the camera orientation.
- Inner-boundary Tracing.
- Acquiring shape information from the Fourier Descriptors.
- Acquiring shape information from the spatial boundary intersection of the leaves.
- Evaluation of the shape data in a Tree-Search scheme and prediction of the original sequence.

To perform these processes a stand-alone software running on Windows OS was developed in C/C++ programming language.

3.1. Image Acquisition

The digital images of the leaves were acquired with a Sony DSC-F505 Cybershot CCD camera in the Museum Rietberg Zurich. The image size was 1600x1200 pixels. Some of the technical settings of the camera at the image acquisition time are given below:

- Image size : 1600x1200 pixels
- Auto Focus : ON
- Focal length : 7.1 mm (at the widest angle position)
- JPEG comp. Quality : Standard

Before the imaging process, three leaves were placed on an A4-sized white paper. Therefore three of the leaves were imaged in per image. Totally 44 images were taken, 22 of them are for front sides and other 22 of them are for back sides.

3.2. Rectification of the Images

Due to the varying orientation of the camera, perspective differences among the images occurred. Especially, the perspective effect due to the unstable *Omega* angle was observed in most images (Figure 2). In order to compensate this effect, *projective transformation* and bi-*linear resampling* processes were applied to each image. In this transformation, the corners of the A4-sized white page were used as reference points.

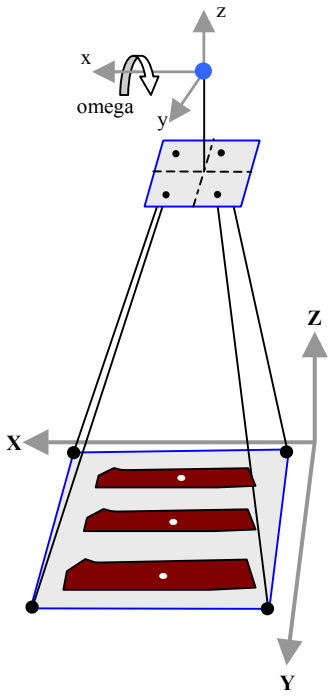
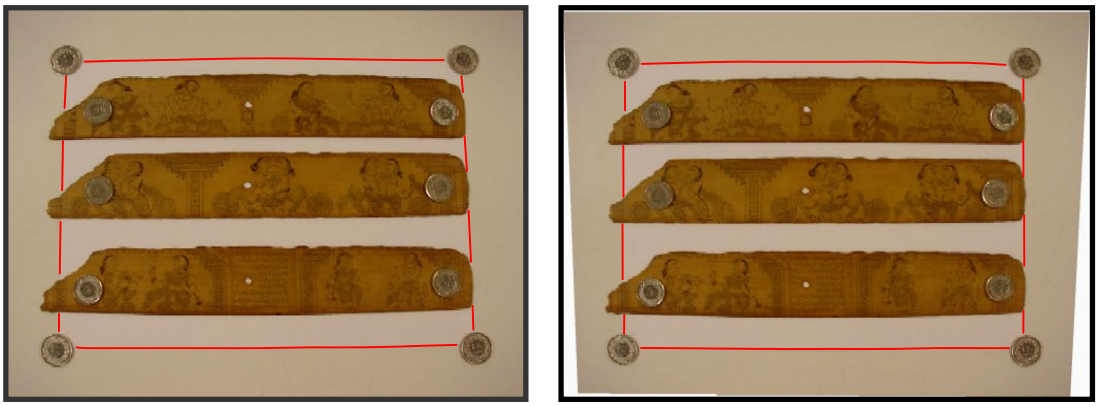


Figure 2: Imaging geometry and common points used in projective transformation

Projective transformation parameters (a_i) were calculated using four common points according to the well known formula (Equation 1). Then, the bi-linear resampling process was applied to obtain the corrected images (Figure 3).

$$X_i = \frac{a_0 \cdot x_i + a_1 \cdot y_i + a_2}{a_6 \cdot x_i + a_7 \cdot y_i + 1} \quad , \quad Y_i = \frac{a_3 \cdot x_i + a_4 \cdot y_i + a_5}{a_6 \cdot x_i + a_7 \cdot y_i + 1} \qquad i = 1,2,3,4 \qquad (1)$$

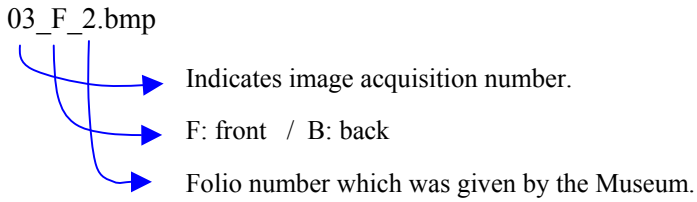


(a) (b)

Figure 3: (a) raw image, (b) projectively transformed image.

Radial distortion of the CCD camera was neglected because it is not relevant according to the working scale.

After the projective transformation, every leaf in the images was cropped and saved as an image file. In the cropping process, the holes in the center of the leaves were assumed as the origin of the predefined 2D coordinate system. This approach provides to lie the all leaves in the same alignment. The file name of the images was given according to the following rule:



3.3. Boundary Tracing

Segmentation is one of the most important image analysis tasks. Its main goal is to delineate the certain objects in the image and to distinguish them from irrelevant image part. There are different segmentation methods according to their search strategies: thresholding techniques, edge based methods, region based methods, and hybrid methods.

Edge based methods commonly use *edge detection operators* such as *Laplacian* (Berzins, 1984), *Sobel* (Sobel, 1970), *Kirsch* (Kirsch, 1971), *Marr-Hildreth* (Marr and Hildreth, 1980), and *Canny* (Canny, 1986) operators. This is usually followed by other processing steps in order to combine edges into edge chains, for example *Hough transformation* (Hough, 1962, Duda and Hart, 1972). Some of the edge based segmentation methods need prior information about shape, such as *Snakes* (Kass et al., 1987), graph searching or dynamic programming based *edge following* methods (Martelli, 1972, Furst 1986). A successful application of LSB-Snakes and dynamic programming for road extraction was given by Gruen and Li (1997). B*oundary tracing* is another well-known edge based segmentation method used to delineate closed shapes (Liow, 1991, Kovalevsky, 1992).

To generate the boundary-based descriptors, boundaries of the every leaf must be delineated. Because of simplicity and suitability to our case, *inner boundary tracing* method was adopted to segment the leaves from image background. An algorithm for inner boundary tracing (Sonka, Hlavac, Boyle, 1993) was implemented (Figure 4). It uses 4-connectivity and 8-connectivity modes. The 8-connectivitiy mode generates fewer points than the 4-connectivity mode. Therefore to implement fast algorithms, the 8-connectivitiy inner boundary-tracing algorithm was applied to all leaf images to generate their boundary (Figure 6). Approximately upper right corner of the all leaves was selected as starting point for the boundary-tracing algorithm.

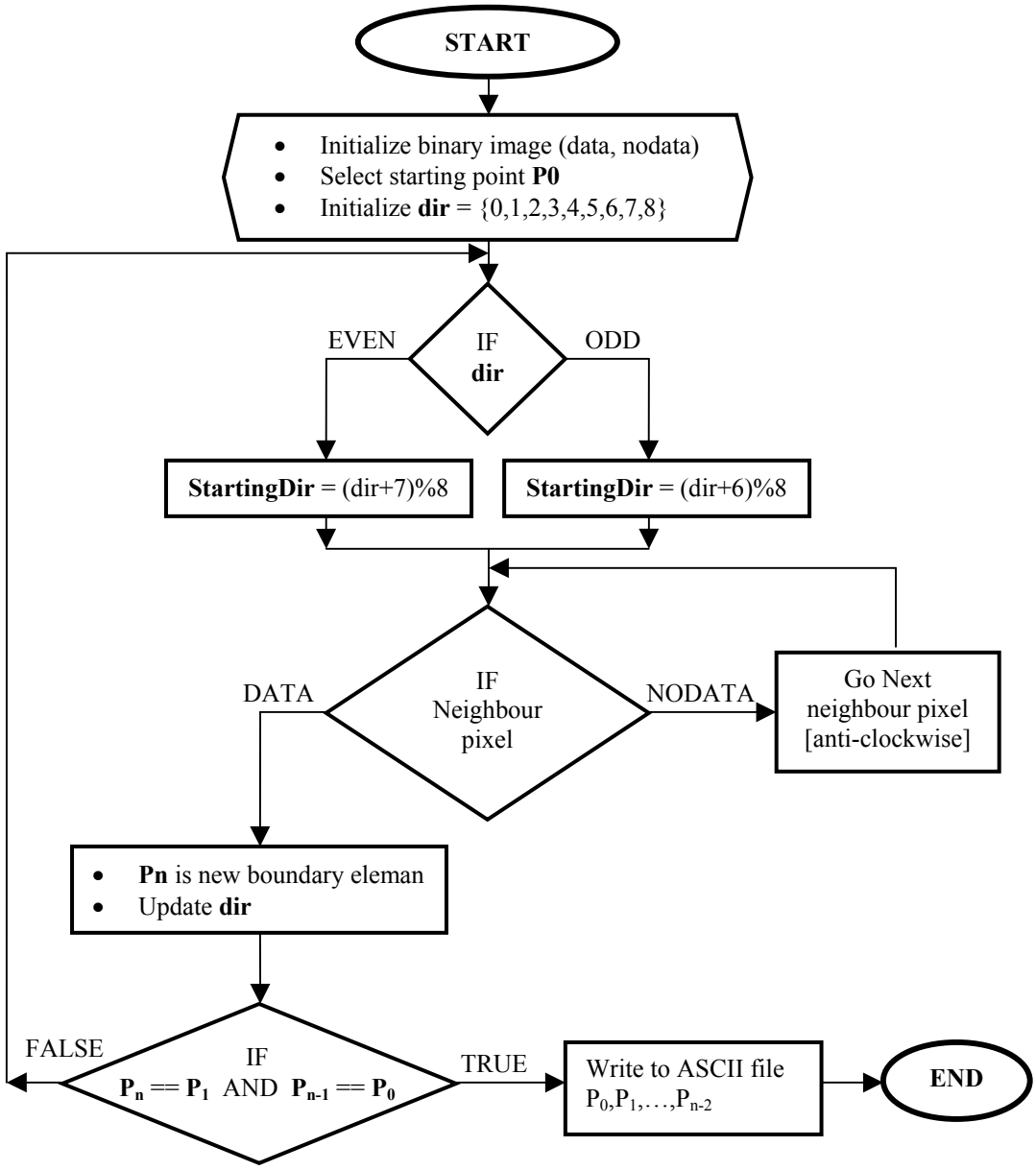


Figure 4: Flow chart of the implemented inner boundary-tracing algorithm.

Implementation details of the applied *inner boundary-tracing algorithm* is given below:

- (1) Select the threshold value for data and no-data gray level regions.
- (2) Get the binary image from original image using threshold value. Put the binary image to a memory array.

  24-bit / 8-bit color / 8-bit gray level => 1 bit (1:data / 0:no-data)

- (3) Select a starting point (**P0**) on the original image and in the mean time also on the background memory array. Pixel **P0** is a starting pixel of the region border.

Define a variable **dir** that stores the direction of the previous move along the border from the previous border element to the current border element.

Assign => **dir = {0,1,2,3,4,5,6,7}** (initialization)

- (4) Search the **3x3** neighborhood of the current pixel in an anti-clockwise direction (Figure 5), but begin from the direction which is formulated in the below:

startingDirection = **(dir+7) mod 8** if **dir** is even

startingDirection = **(dir+6) mod 8** if **dir** is odd

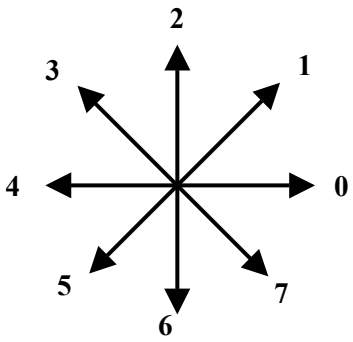


Figure 5: Direction codes.

The first pixel that is found with data-code is a new boundary element of **Pn.**

Update the **dir** value.

- (5) If the current boundary element **Pn** is equal to the second border element **P1**, and if the previous border element **Pn-1** is equal to **P0**, stop. Otherwise repeat step (4).
- (6) The detected inner border is represented by pixels **P0 … Pn-2**.

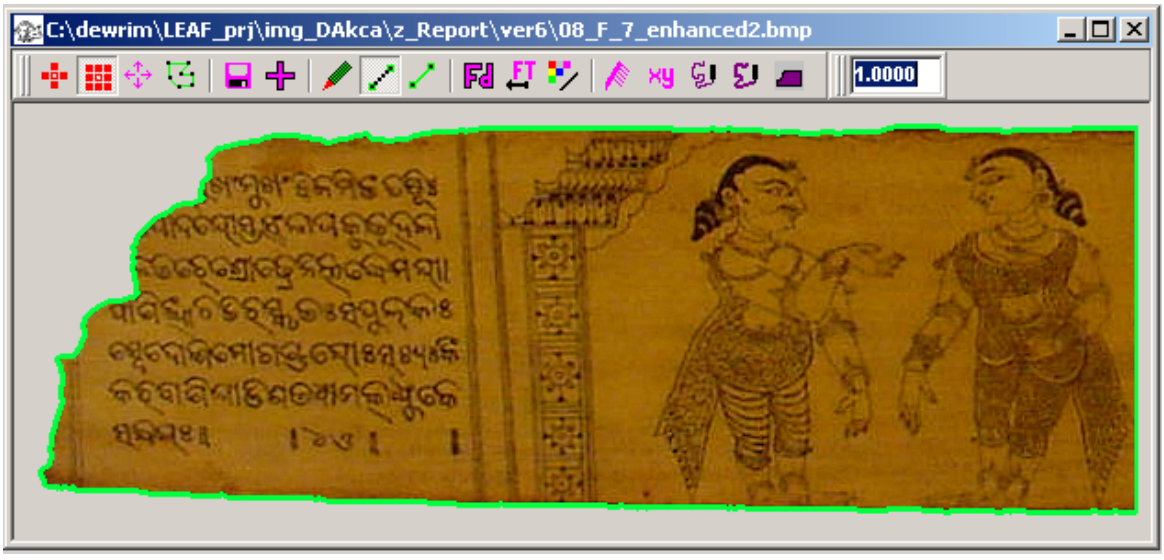


Figure 6: An example of the boundary-tracing algorithm.

After the operation, inner boundary coordinates of every leaf were saved to an ASCII file. The developed software can generate boundary coordinates of the leaf images in a semi-automatic or fully automatic manner.

3.4. Fourier Descriptors

Fourier descriptors method is a 2-D boundary analysis method, is based on Fourier analysis of the function derived from the boundary (Richard and Hemami, 1974, Zhan and Roskies, 1972, Persoon and Fu, 1977). Main advantage of the method is invariance to translation, rotation and scaling of the shape to be described. Therefore, shape description becomes independent of the relative position and size of the object in the image space.

One of the most popular implementation area of the Fourier descriptors is handwritten character recognition (Granlund, 1972, Cao et al., 1994). Classification of 2-D partial shapes using Fourier descriptors was given by Lin and Chellapa (1986). Another interesting implementation of Fourier descriptors in Medicine was presented by Veropoulos et al. (1998). In this work, boundary tracing followed by Fourier shape descriptors was used to represent Tubercle Bacilli. In photogrammetry discipline, matching of the area and line features with Fourier descriptors was proposed as a feature-based matching method (Tseng and Schenk, 1992, Tseng et al., 1997).

Let us represent the boundary points as coordinate pairs; (x_0,y_0), (x_1,y_1), ……. , (x_{N-1},y_{N-1}). Each coordinate pair can also be represented as a complex number. In this case, the x axis is treated as the real axis and the y axis as the imaginary axis of a sequence of complex numbers.

$$s(k) = x(k) + j.y(k) \qquad k=0,1,2,\ldots,N-1 \qquad (2)$$

This representation has one great advantage: It reduces a 2D to a 1D problem. The discrete 1D Fourier transform of *s(k)* is:

$$a(u) = \frac{1}{N}\sum_{k=0}^{N-1} s(k).\exp\left[-j.2.\pi.u.k/N\right] \qquad u= 0,1,2,\ldots,N-1 \qquad (3)$$

The complex coefficients $a(u)$ are called the *Fourier descriptors* of the boundary. The inverse Fourier transform of the $a(u)$'s restores $s(k)$.

$$s(k) = \sum_{u=0}^{N-1} a(u).\exp\left[j.2.\pi.u.k/N\right] \qquad k= 0,1,2,\ldots,N-1 \qquad (4)$$

Instead of full set of $a(u)$.. coefficients only first M coefficients can also be used in the inverse transformation. The result is an approximation of $s(k)$.

$$\tilde{s}(k) = \sum_{u=0}^{M-1} a(u).\exp[j.2.\pi.u.k / N] \qquad k= 0,1,2,\ldots,N-1 \quad , \quad M<N \qquad (5)$$

That is, the same number of points exist in the approximate boundary, but not as many terms are used in the reconstruction of each point. High frequency components represent fine details, and low frequency components determine the global shape. Thus the smaller M becomes, the more detail is lost on the boundary (Figure 7).

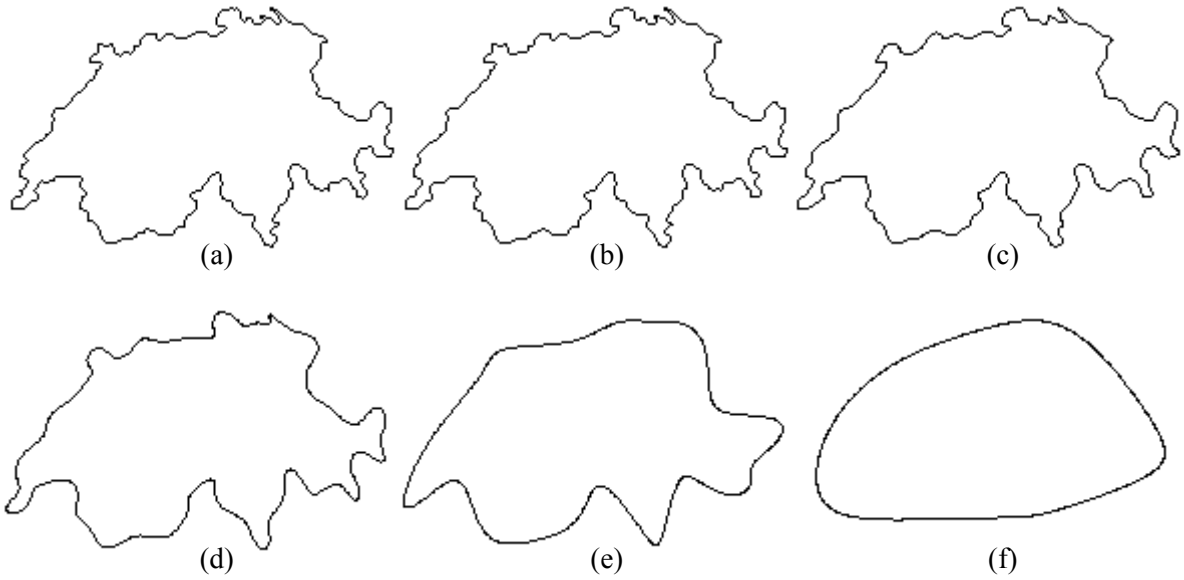


Figure 7: (a) original boundary has 708 boundary elements, (b), (c), (d), (e), and (f) are inverse Fourier transforms used 250, 150, 75, 25, and 10 coefficients respectively.

A few Fourier descriptors can be used to capture the gross essence of a boundary. This property is valuable, because these coefficients carry shape information. Thus they can be used as the basis for differentiating between distinct boundary shapes (Gonzales and Woods, 1993).

The most known property of the Fourier descriptors is that only a few percent of the descriptors can represent the whole boundary, as mentioned before. This property can also be observed in our leaf examples. In Figure (8), there are 1532 boundary elements. To capture the whole shape, only 250 Fourier coefficients (~ %16) were used in reverse Fourier transformation.

In this application, Fourier descriptors of the boundary elements were calculated using 1D Fast Fourier transformation. Then, Euclidean distance between the first 250 Fourier descriptors of the two different leaves is used as a shape similarity measure. The number of Fourier descriptors (250) used for calculation of the shape similarity measure was determined empirically.

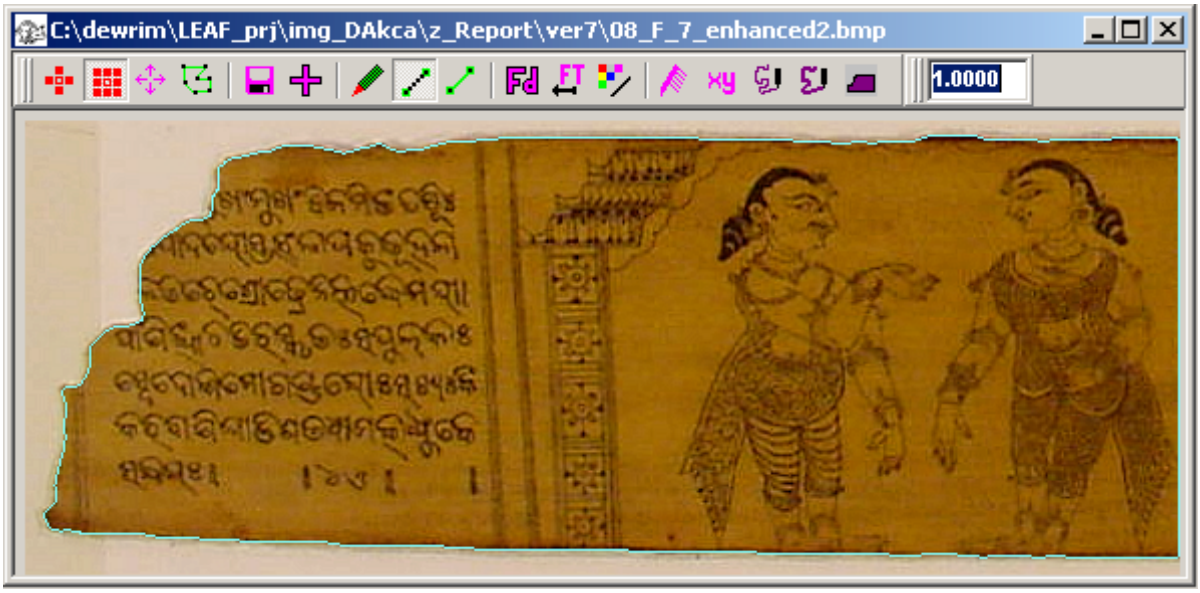


Figure 8: Original boundary has 1532 boundary elements. 250 Fourier coefficients were used to capture the whole leaf boundary.

Also, these distance measures are aided by additional shape features (**area** and **dMax**) with proper weights (Figure 9).

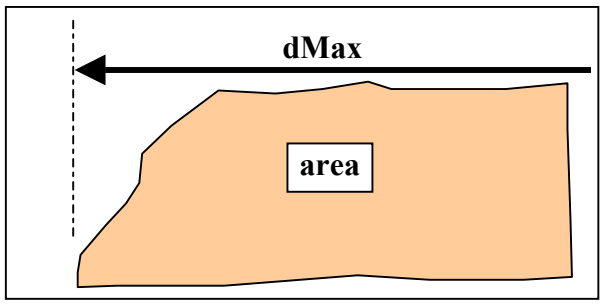


Figure 9: Additional shape features.

Finally Euclidean distance between the two vectors is calculated as below formula:

$$v_{ij} = \begin{bmatrix} a(u_0)_i - a(u_0)_j \\ a(u_1)_i - a(u_1)_j \\ \cdots \\ a(u_{249})_i - a(u_{249})_j \\ dMax_i - dMax_j \\ area_i - area_j \end{bmatrix}_{252x1} \qquad P = \begin{bmatrix} 1 & 0 & .. & 0 & 0 & 0 \\ 0 & 1 & .. & 0 & 0 & 0 \\ .. & .. & .. & .. & .. & .. \\ 0 & 0 & .. & 1 & 0 & 0 \\ 0 & 0 & .. & 0 & W_{dMax} & 0 \\ 0 & 0 & .. & 0 & 0 & W_{area} \end{bmatrix}_{252x252} \tag{6}$$

W_{dMax} and W_{area} weight values were determined empirically (W_{dMax}= 0.1 , W_{area}= 0.001).

$$d_{ij} = \sqrt{v^T \cdot P \cdot v} \qquad i = 0,1,...,65 \quad , \quad j = 0,1,...,65 \quad , \quad i \neq j \tag{7}$$

d_{ij} : Euclidean distance between the i_{th} and j_{th} leaves.

All Euclidean distances among the leaves were calculated, and an overall symmetric distance matrix was generated.

$$D_{fourier} = \begin{bmatrix} 0.0 & d_{0,1} & .. & d_{0,65} \\ d_{1,0} & 0.0 & .. & d_{1,65} \\ .. & .. & .. & .. \\ d_{65,0} & d_{65,1} & .. & 0.0 \end{bmatrix}_{66x66} \tag{8}$$

On the $D_{fourier}$ matrix one can determine which leaves are similar. In the data evaluation step, this matrix was used to find the sequence.

3.5. Spatial Boundary Intersection

Another shape similarity measure apart from the Fourier descriptors was also used. In this approach, boundaries of every different two leaves were intersected (Figure 10), and two different related features upon the intersected area were calculated: *intersected area* and *standard deviation of the inner distances* (in pixel unit).

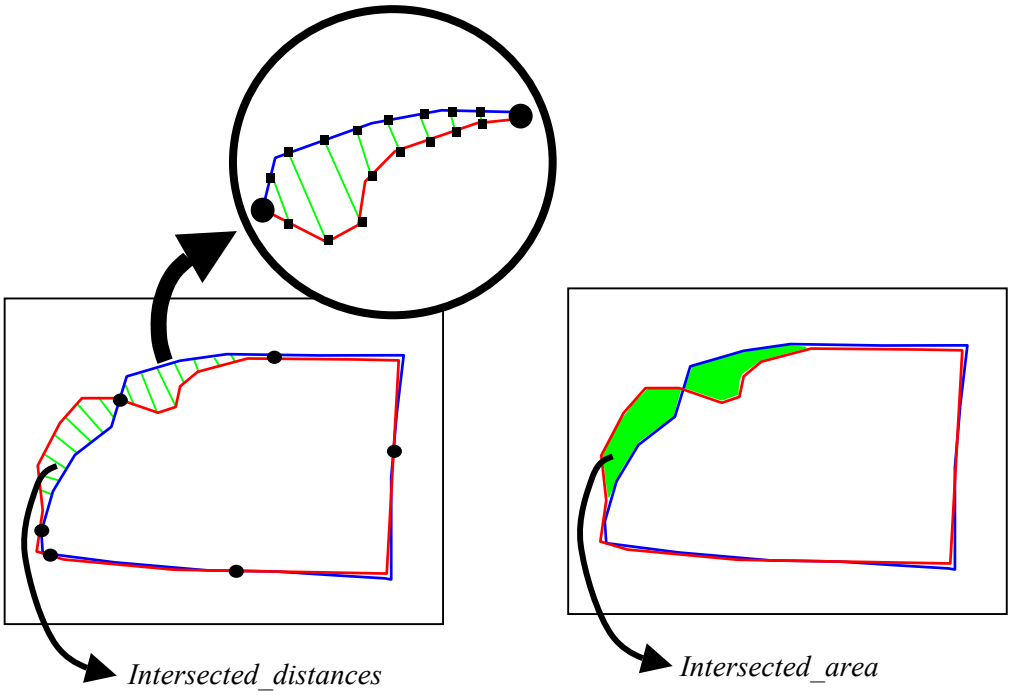


Figure 10: Standard deviation of the inner distances and intersected area.

Standard deviation of the inner distances was calculated according to Equation (9). Start and end points of the inner lines were determined according to junction points of two boundaries (Figure 10). Equally spaced points were determined along the both boundary part between the two junction points, and these points were linked as inner lines.

$$\text{int\_dist} = \sqrt{\frac{\sum \Delta d^2}{n}} \qquad \text{n: number of lines.} \tag{9}$$

Since cropping of individual leaf images followed by the rectification process was carried out in a common coordinate system that is defined by the holes in center of every leaf, all leaves have same alignment.

These two different shape descriptors were merged as one value with using proper weights (Equation 10).

$$dI_{ij} = \text{int\_dist}_{ij} \times W_{\text{int\_dist}} + \text{int\_area}_{ij} \times W_{\text{int\_area}} \qquad i \neq j \tag{10}$$

W_{int_dist} and W_{int_area} weight values were determined empirically (W_{int_dist}=1.0 , W_{int_area}=0.002).

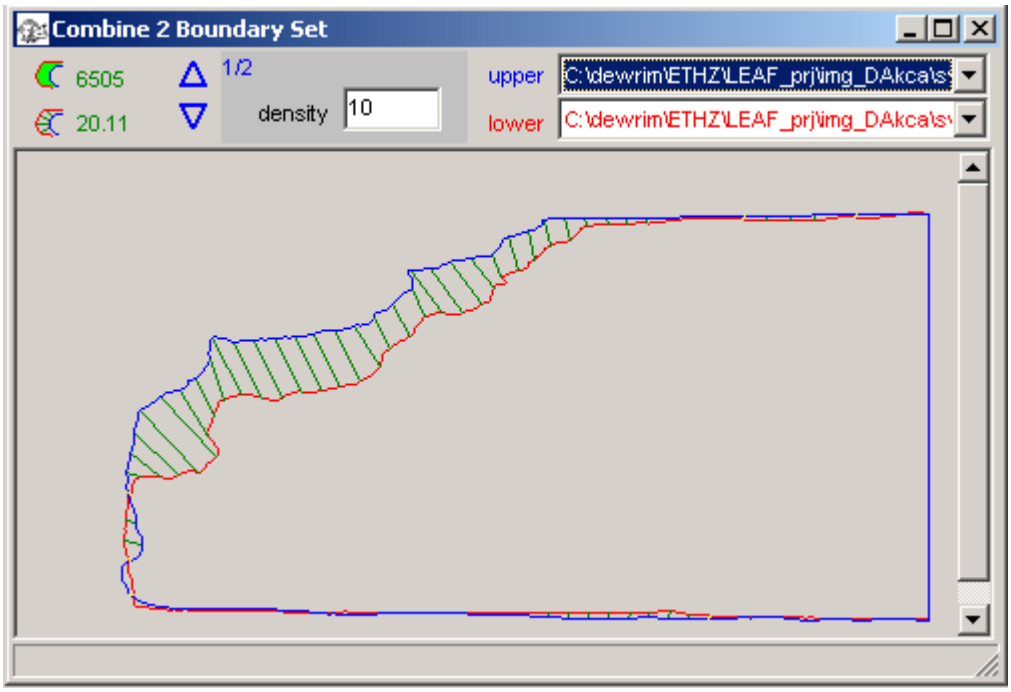


Figure 11: An example of the spatial boundary intersection.

All $\mathbf{dI_{ij}}$ values among the leaves were calculated, and an overall matrix was generated.

$$D_{intersection} = \begin{bmatrix} 0.0 & dI_{0,1} & .. & dI_{0,65} \\ dI_{1,0} & 0.0 & .. & dI_{1,65} \\ .. & .. & .. & .. \\ dI_{65,0} & dI_{65,1} & .. & 0.0 \end{bmatrix}_{66x66} \quad (11)$$

Similar to $D_{fourier}$ matrix this matrix also gives useful metric information about how they are similar according to their shape for an arbitrary selected leaf pair. In the processing steps it was observed that both matrices ($D_{fourier}$ and $D_{intersection}$) give very similar results.

3.6. Evaluation of the Shape Data using Tree Search

In the evaluation step, $D_{fourier}$ and $D_{intersection}$ matrices were used to generate the proposed sequence. These matrices can easily solve the *partial problems:* which leaves might be ancestor and successor for a pointed leaf in the sequence? But the *global problem* still remains: how can this information be evaluated efficiently in order to generate full sequence?

In the first attempt, a simple evaluation scheme (Figure 12) was designed as follows:

- (1) Define a starting leaf.
- (2) Find the most similar leaf to the present leaf on the matrix D.

  Query whether the target leaf was used at a previous step. If it is YES, go to next similar leaf which is pointed on matrix D. If it is NO, Go to the target leaf, and assign it as present leaf.
- (3) Check the sequence. If it is finished, exit. Otherwise, go to step (2).

Since the first 18 leaves were already fixed by the Museum, the algorithm started from the 18th leaf. Although it provides good results for the first 1st – 3rd quarter of the full sequence, some of the last leaves are slightly different with respect to their neighbour leaves on the sequence. The reason of the outcoming result was the one-way search strategy of the implementation.

In order to develop a better evaluation scheme, *tree search* method was adopted. One of the well known application topic of tree search methods is *relational matching*. The more general matching scheme has been developed by computer vision researchers (Shapiro and Haralick 1987, Boyer and Kak 1988). Successful applications of relational matching in photogrammetry were given by Haala and Vosselman (1992), Zilberstein (1992), Cho (1996), and Wang (1996). Several examples of potential applications of tree search methods in digital photogrammetry were also given by Vosselman (1995).

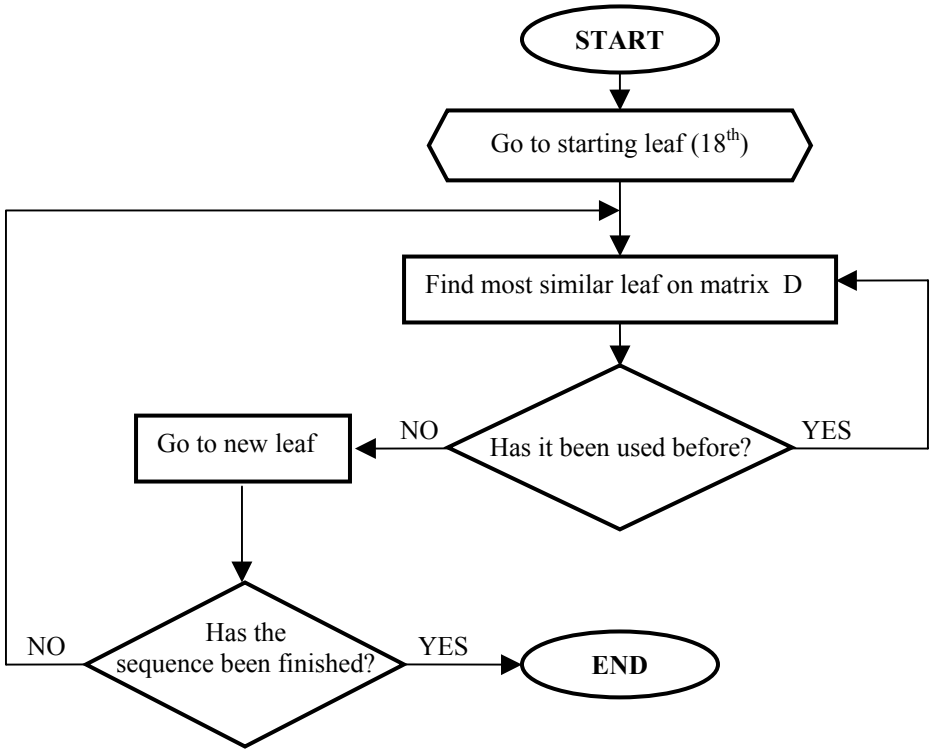


Figure 12: Implemented simple evaluation scheme

Search *trees* consist of hierarchical *nodes* connected by *arcs*. Tree starts from a root node and descends into successor nodes in a branched structure. An ancestor node can only branch into possible successor nodes in order to keep the volume of tree in acceptable limits. In the case of relational matching, the problem is to match two primitive sets, namely relational descriptions. The primitives, let us say $p_i = \{p_1 , p_2 , \ldots, p_n\}$ $p_i \in P$, of one relational description are called *units,* and the primitives of the description to be matched, $q_i = \{q_1 , q_2 , \ldots, q_m\}$ $q_i \in Q$, are termed *labels*. The number of units defines the depth of the tree. The best mapping of $P \rightarrow Q$ is the path with lowest cost.

In our example a tree search scheme that starts from the 18^{th} (fixed) leaf and ends at the 66^{th} (relaxed) leaf was established. The rings of the sequence chain are defined as *units.* Therefore, number of *units*, namely depth of the tree, and number of *labels* are the same (49). Every node in the tree was defined as a leaf and branched to the most probable neighbour leaves. The root node is the 18^{th} leaf, which is the last leaf of the page numbered leaves. The similarity measures, calculated using shape descriptors, were expressed as the costs of the arcs, which connect two nodes in the tree. Implemented tree search method proposes the total path with minimum cost as the most probable sequence. Figure 13 shows a part of the designed tree search.

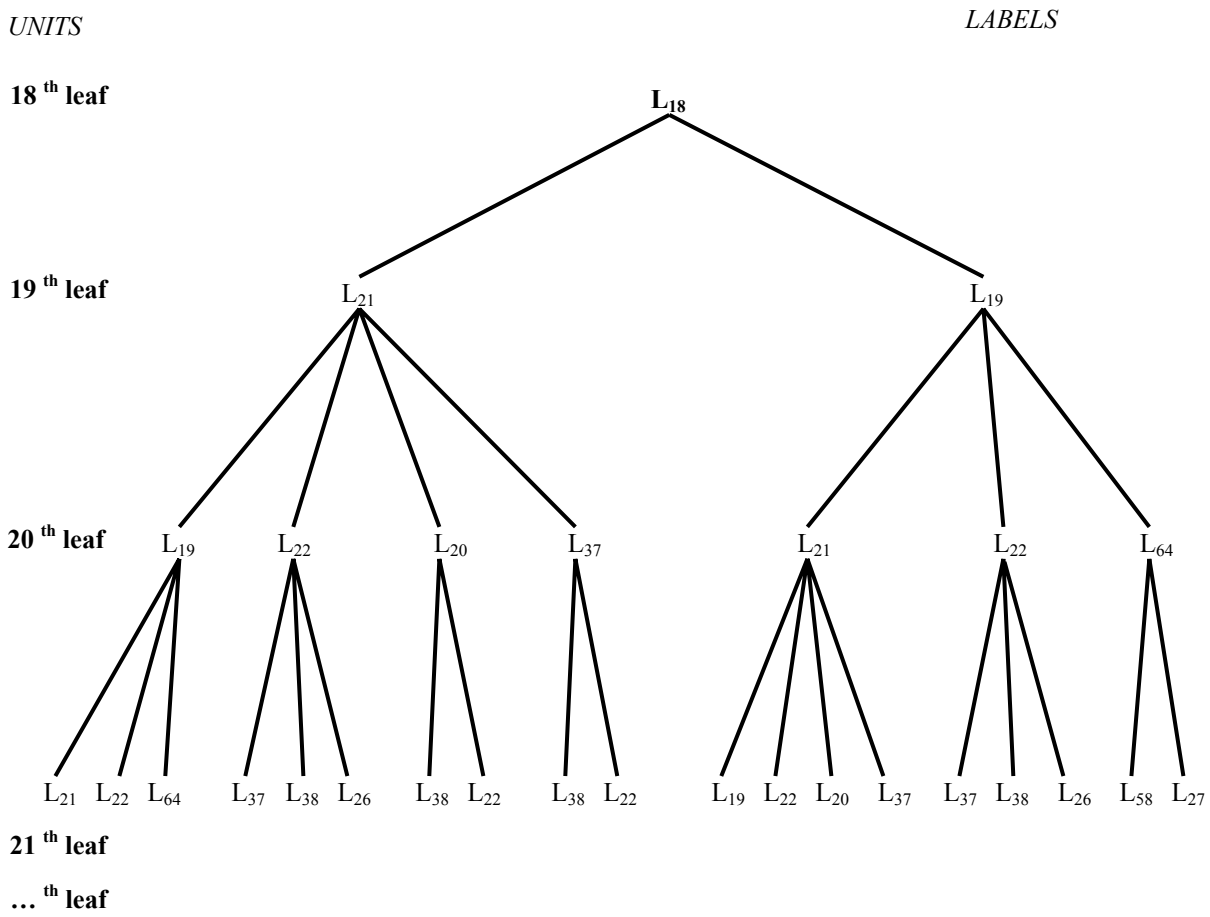


Figure 13: A sample part from the implemented tree search (L_j indicates *leaf j*)

In the implementation phase, number of allowed successor leaves for an ancestor leaf was defined according to two predefined rules (limits):

- Limit T_c : maximum count for allowed nodes
- Limit T_v : maximum value limit for shape similarity measure from matrix D_{ij}

These two limits state that the number of successor nodes for an ancestor node can not exceed *limit* T_c , and the cost value for any arc can not exceed *limit* T_v . In tree search methods, there is a tradeoff between exponential increasing search time and searching all possible combinations. These two limits compensate the mentioned tradeoff.

Some leaves in the set have completely different shapes with respect to other leaves. This causes either none of the leaf might not give a branch (arc) to these different shaped leaves or these extremely different leaves might not find similar leaf (leaves) as successor. To get rid of this problem one should relax the limits T_c and T_v . In this case, the search time increases exponentially. The leaves are splitted into groups to solve the problem in an optimum way. According to their shapes, the leaves were grouped into four categories. Each group was ranked internally and then all the groups were merged.

4. CONCLUSIONS

In this work, the missing sequence of the historical palm leaf manuscript was treated to find using shape descriptors. Two boundary-based shape descriptor algorithms were applied to the boundary data. Both of the shape description algorithms can find the most similar neighbour leaf for a pointed leaf successfully.

The fundamental problem of this work is to establish the original sequence using shape similarity information provided by the shape description algorithms. A tree search scheme was performed to achieve this task. The sequence, which has the minimum total cost, was proposed as the most probable sequence.

An alternative method to the tree search method might be probabilistic relaxation method. In this case, similarity measures provided by the shape description algorithms can be interpreted as probability of neighborhood.

In order to provide visual inspection for the proposed sequence, texture mapped model of palm leaf manuscript was visualized and animated.

ACKNOWLEDGMENTS

This project was supported by Museum Rietberg, Zuerich. Cooperation and help of Dr. Eberhard Fischer is gratefully acknowledged.

5. REFERENCES

- Berzins, V., 1984. Accuracy of Laplacian edge detectors. Computer Vision, Graphics, and Image Processing, vol.27, pp. 1955-2010.
- Boyer, K.L., and Kak, A.C., 1988. Structural stereopsis for 3-D vision. IEEE Transactions on Pattern Analysis and Machine Intelligence, 10(2), pp. 144-166.
- Canny, J.F., 1986. A computational approach to edge detection. IEEE Transactions on Pattern Analysis and Machine Intelligence, 8(6), pp. 679-698.
- Cao, J., Ahmadi, M., and Shridhar, M., 1994. Handwritten numeral recognition with multiple features and multistage classifiers. In International Symposium on Circuits and Systems, May 30- June 2, vol.6, pp.323-326.
- Cho, W., 1996. Relational matching for automatic orientation. International Archives of Photogrammetry and Remote Sensing, XXXI (B3), pp.111-119.
- Duda, R.O., Hart, P.E., 1972. Using the Hough transforms to detect lines and curves in pictures. Communications of the ACM, 15(1), pp. 11-15.
- Fischer, E., 2002. Fruehe Sanskrit-Liebesgedichte. In: Liebeskunst, Liebeslust und Liebesleid in der Weltkunst, Museum Rietberg Zuerich, pp. 154-162.

- Furst, M.A., 1986. Edge detection with image enhancement via dynamic programming. Computer Vision, Graphics, and Image Processing, vol.33, pp.263-279.
- Gonzalez, R.,C., and Woods, R.,E., 1993. Digital Image Processing. Addison-Wesley Publishing, USA.
- Granlund, G.H., 1972. Fourier preprocessing for hand printed character recognition. IEEE Transactions on Computers, vol. C-21, pp. 195-201.
- Gruen, A., and Li, H., 1997. Semi-automatic linear feature extraction by dynamic programming and LSB-Snakes. Photogrammetric Engineering & Remote Sensing, 63(8), pp. 985-995.
- Haala, N., and Vosselman, G., 1992. Recognition of road and river patterns by relational matching. International Archives of Photogrammetry and Remote Sensing, XXIX (B3), pp. 969-975.
- Hough, P.V.C., 1962. A method and means for recognizing complex patterns. U.S. Patent 3,069,654.
- Kass, M., Witkin, A., and Terzopoulos, D., 1987. Snakes: Active contour models. In Proceedings, First International Conference on Computer Vision, 8-11 June, IEEE, London, pp.259-268.
- Kirsch, R., 1971. Computer determination of the constituent structure. Computers and Biomedical Research, vol.4, pp. 315-328.
- Kovalevsky, V., 1992. Finite topology and image analysis. In: Hawkes, P. (ed.), Advances in Electronics and Electron Physics, vol.84, Academic Press, pp. 197-259.
- Lin, C.C., and Chellapa, R., 1986. Classification of partial 2-D shapes using Fourier descriptors. In IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 22-26 June, Florida, pp. 344-350.
- Liow, Y.T., 1991. A contour tracing algorithm that preserves common boundaries between regions. CVGIP – Image Understanding, 53(3), pp.313-321.
- Marr, D., Hildreth, E.C., 1980. Theory of edge detection. Proceedings of the Royal Society of London, vol.207, pp. 187-217.
- Martelli, A., 1972. Edge detection using heuristic search methods. Computer Vision and Image Processing, vol.1, pp. 169-182.
- Persoon, E., and Fu, K.S., 1977. Shape discrimination using Fourier descriptors. IEEE Transactions on System, Man, and Cybernetics, vol. SMC-7, pp. 170-179.
- Richard, C.W., and Hemami, H., 1974. Identification of three-dimensional objects using Fourier descriptors of the boundary curve. IEEE Transactions on System, Man, and Cybernetics, vol. SMC-4, pp. 371-378.
- Shapiro, L.H., and Haralick, R.M., 1987. Relational matching. Applied Optics, vol. 26, pp. 1845-1851.
- Sobel, I., 1970. Camera models and machine perception. AIM-21, Stanford Artificial Intelligence Lab, Palo Alto.
- Sonka, M., Hlavac, V., and Boyle, R., 1993. Image Processing, Analysis, and Machine Vision. Chapman&Hall Computing, London.
- Tseng, Y.H., and Schenk, T., 1992. A least squares approach to matching lines with Fourier descriptors. International Archives of Photogrammetry and Remote Sensing, XXIX (B3), pp. 469-475.

- Tseng, Y.H., Tzen, J.J., Tang, K.P., and Lin, S.H., 1997. Image-to-image registration by matching area features using Fourier descriptors and Neural Networks. Photogrammetric Engineering & Remote Sensing, 63(8), pp. 975-983.
- Veropoulos, K., Learmonth, G., Campbell, C., Knight, B., and Simpson, J., 1998. Automated identification of Tubercle Bacilli in Sputum. In Proceedings of the 8th International Conference on Artificial Neural Networks (ICANN 98), vol.2, 2-4 September, Skoerde, Sweden, Springer, pp. 277-281.
- Vosselman, G., 1995. Applications of tree search methods in digital photogrammetry. ISPRS Journal of Photogrammetry and Remote Sensing, 50(4), pp. 29-37.
- Wang, Y., 1996. Structural matching and its applications for photogrammetric automation. International Archives of Photogrammetry and Remote Sensing, XXXI (B3), pp. 918-923.
- Zhan, C.T., and Roskies, R.S., 1972. Fourier descriptors for plane closed curves. IEEE Transactions on Computers, vol. C-21, pp.269-281.
- Zilberstein, O., 1992. Relational matching for stereopsis. International Archives of Photogrammetry and Remote Sensing, XXIX (B3), pp. 711-719.

6. APPENDIX

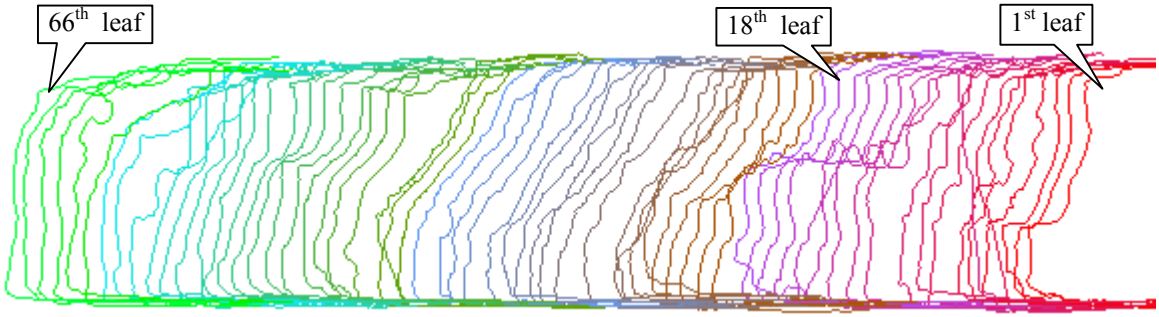


Figure 14: Orthographic view of the proposed sequence (constant shift: 15 pixels).

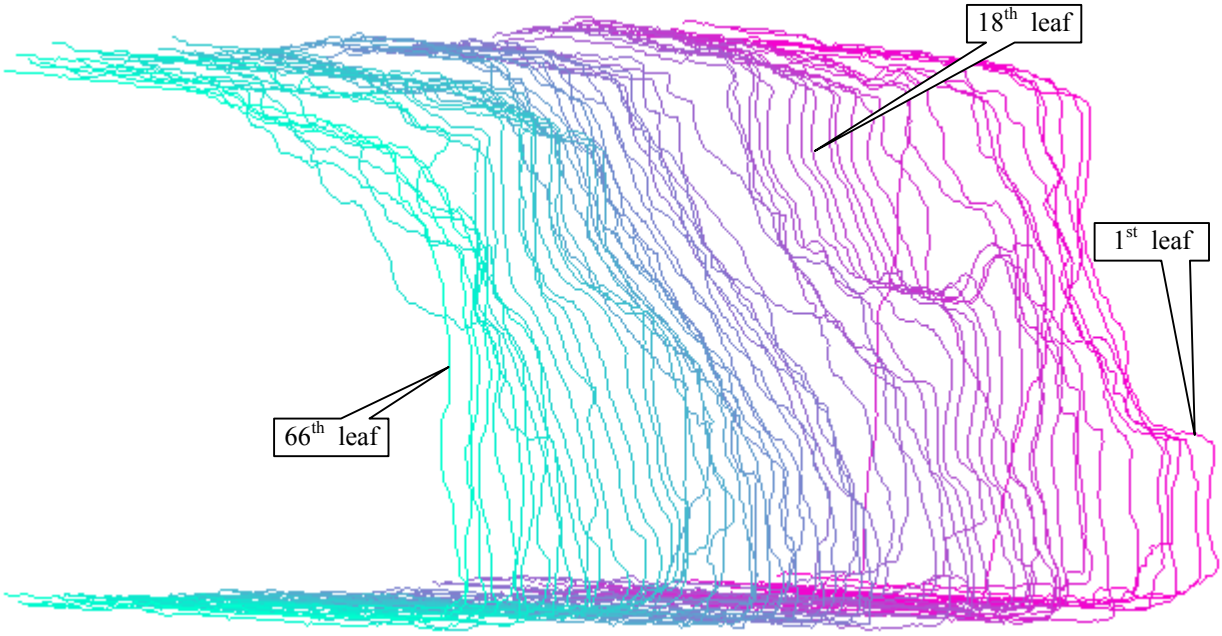


Figure 15: Orthographic view of the proposed sequence (constant shift: 4 pixels).

Figure 16: Texture mapped model.

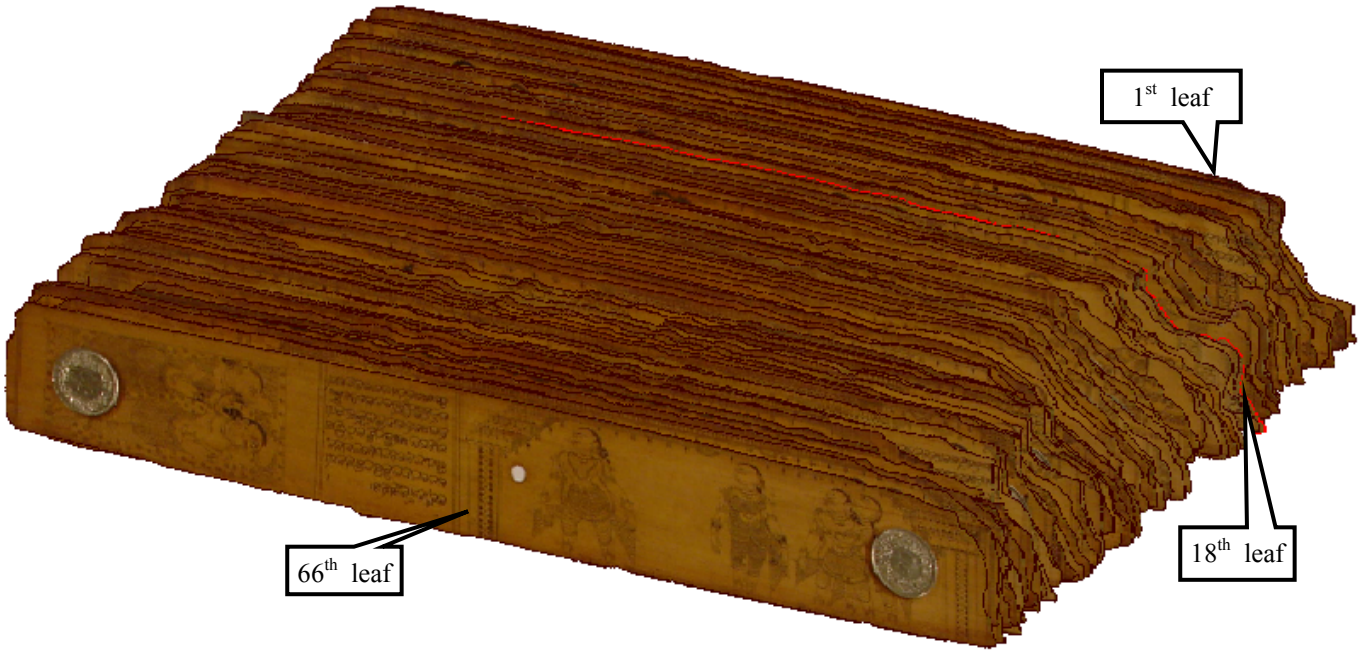


Figure 17: Texture mapped model of the palm leaf manuscript.

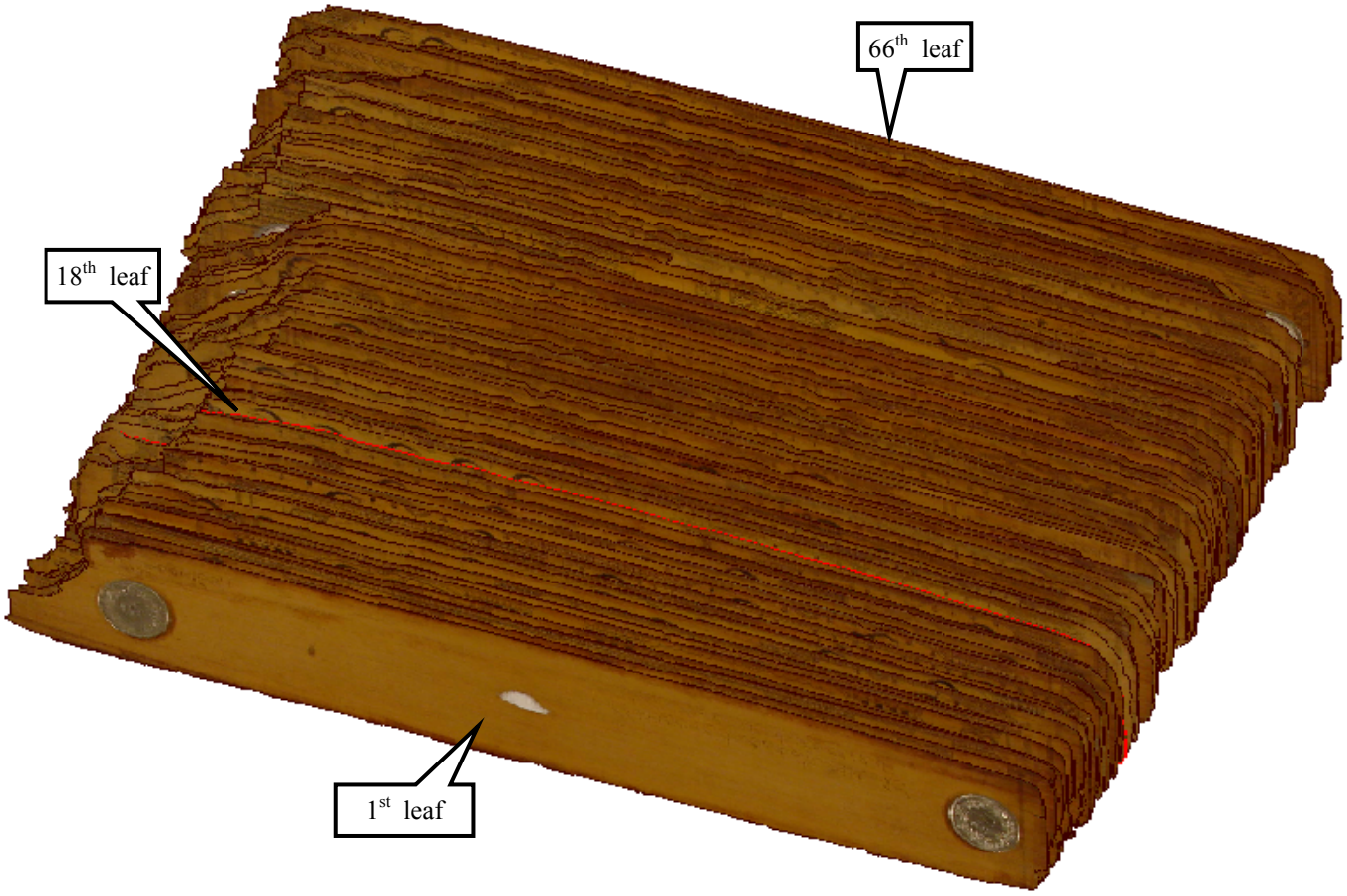


Figure 18: Texture mapped model of the palm leaf manuscript (upper view).


0. \00_F_0.bmp


1. \01_F_1.bmp


2. \02_F_1A_a.bmp


3. \03_F_2_a.bmp

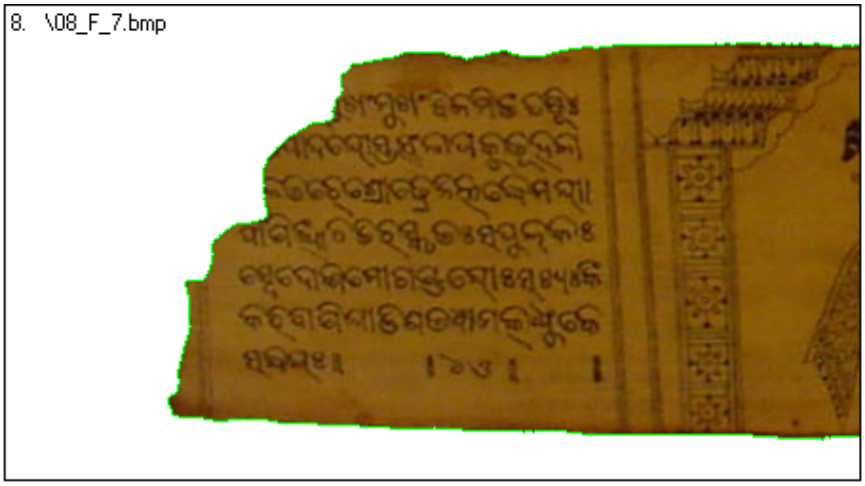
4. \04_F_3.bmp

5. \05_F_4.bmp

6. \06_F_5.bmp

7. \07_F_6.bmp

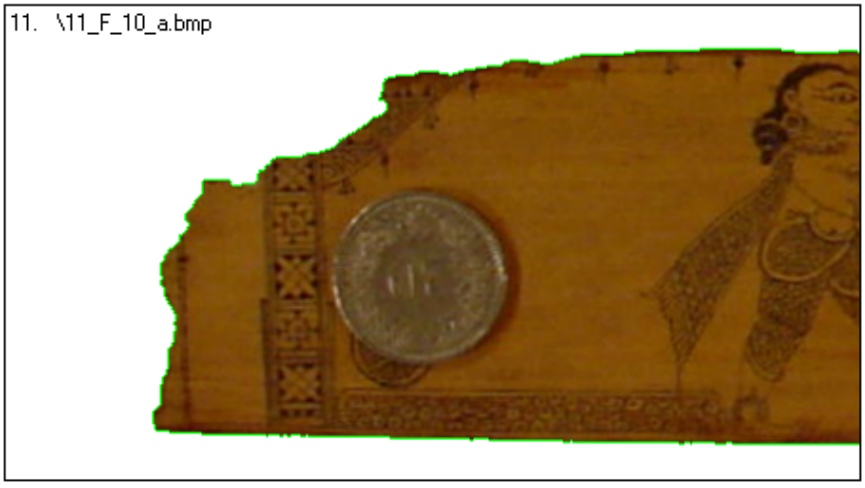
8. \08_F_7.bmp



9. \09_F_8_a.bmp

10. \10_F_9.bmp

11. \11_F_10_a.bmp



12. \12_F_11.bmp

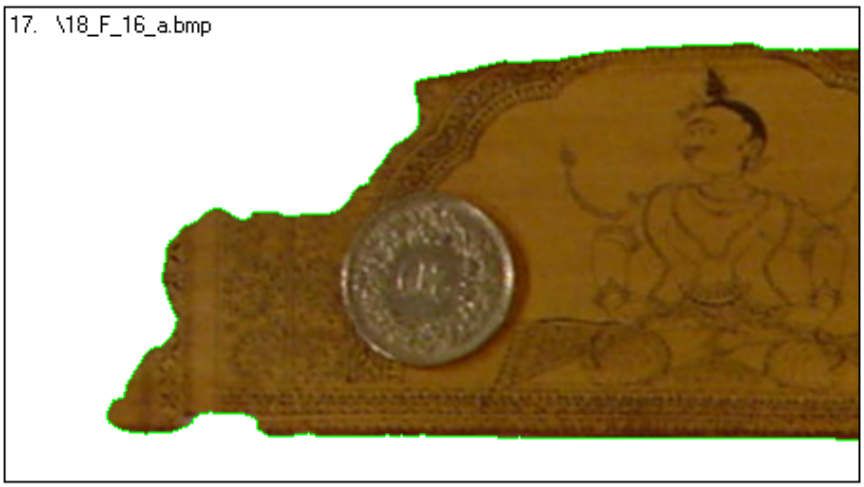
13. \13_F_12.bmp

14. \14_F_13.bmp

15. \15_F_14_a.bmp

16. \17_F_15_a.bmp

17. \18_F_16_a.bmp

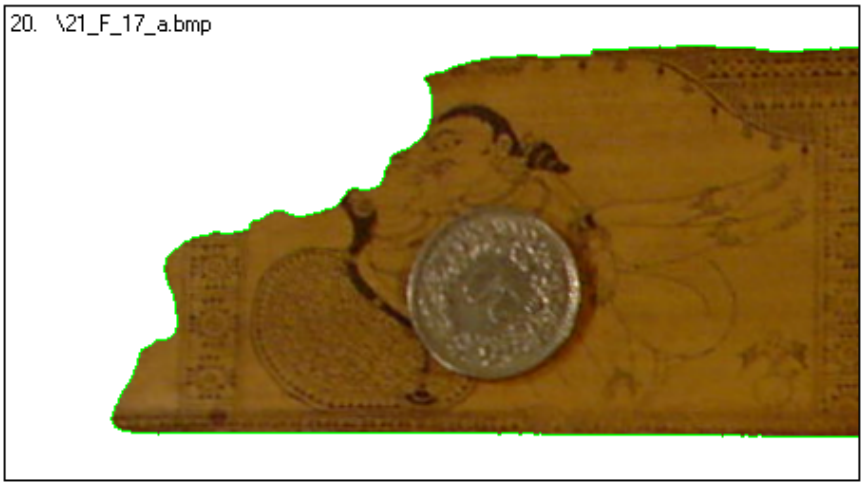


FIXED Leaf
at the beginning

18. \19_F_16A.bmp

19. \20_F_16B.bmp

20. \21_F_17_a.bmp



21. \22_F_18_a.bmp

22. \37_F_32_a.bmp

23. \46_F_40.bmp


24. \38_F_33.bmp


25. \35_F_30.bmp


26. \36_F_31.bmp


27. \29_F_24_a.bmp

28. \42_F_36.bmp

29. \45_F_39.bmp

30. \63_F_J.bmp

31. \54_F_49_a.bmp

32. \28_F_23_a.bmp

33. \24_F_20_a.bmp

34. \31_F_26.bmp

35. \25_F_20A.bmp


36. \23_F_19_a.bmp


37. \26_F_21_a.bmp


38. \56_F_B.bmp


39. \27_F_22_a.bmp

40. \58_F_D.bmp

41. \64_F_K_a.bmp

42. \59_F_E.bmp

43. \57_F_C.bmp

44. \61_F_H.bmp

45. \62_F_I_a.bmp

46. \65_F_L_a.bmp

47. \55_F_50.bmp


48. \50_F_45_a.bmp


49. \52_F_47.bmp


50. \16_F_14A.bmp


51. \49_F_44.bmp

52. \41_F_35A_a.bmp

53. \51_F_46_a.bmp

54. \53_F_48_a.bmp

55. \48_F_43.bmp

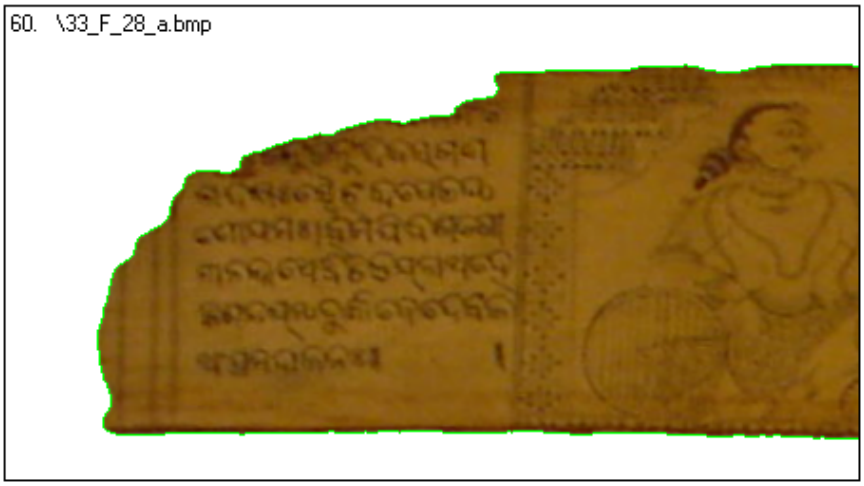
56. \47_F_42_a.bmp

57. \34_F_29.bmp

58. \44_F_38_a.bmp

59. \60_F_F.bmp

60. \33_F_28_a.bmp



61. \40_F_35.bmp

62. \43_F_37_a.bmp

63. \30_F_25_a.bmp


64. \32_F_27_a.bmp


65. \39_F_34.bmp