# Robust Occluding Contour Detection Using the Hausdorff Distance *

Xilin Yi[1] and Octavia I. Camps[1,2]
[1]Department of Electrical Engineering
[2]Department of Computer Science and Engineering
The Pennsylvania State University
University Park, PA, 16802
E-mails: {xilin,camps}@whale.ece.psu.edu

## Abstract

In this paper, a correlational approach for distinguishing occluding contours from object markings for 3D object modeling is presented. The proposed method is valid under weak perspective projection, does not require to search for correspondences between frames, can handle scaling between consecutive images, thus can estimate the full Euclidean surface structure, and does not require camera calibration or camera motion measurement. Extensive experimental results show that the method is robust to the occlusion of feature points and image noise unlike previous affine-based approaches. Qualitative and quantitative results for the relation between the required minimum viewing angle change for the detection and the surface curvature are also presented.

**KEY WORDS:** Occluding contour, object modeling, Hausdorff distance.

## 1 Introduction

Several approaches have been developed for 3D object modeling using silhouettes or occluding contours [1, 3, 4, 11, 15]. Two important issues with previous approaches are the availability of accurate camera motion parameters and the robustness of the occluding contour detection.

Zisserman, Blake and Cipolla, and Giblin [2, 5, 15] used the concept of occluding contour and marking stationarity to distinguish and prior known surface marking points to distinguish the deformed occluding contours from other surface markings. However, a problem with this approach is that if the viewpoint is changed, the prior known marking pattern might be lost, strongly affecting the results. Kutulakos [10] extended their work by enforcing active viewpoint control, and selecting non-coplanar points with their tangents along the viewpoint motion plane as an affine basis to identify the occluding contours. While this approach avoids using prior knowledge of markings, it has the following disadvantages: it works well only under orthographic projection; it requires tracking of all the feature points; and it is sensitive to the occlusion of the affine basis feature points due to its computational approach.

In this paper, we present a correlational approach using the stationarity properties of occluding contours and markings that eliminates most of the above disadvantages. The approach does not require knowledge of the camera motion parameters, and does not need surface markings to be known *a priori* either. We first actively control the camera motion plane, and select tangent points along the camera motion plane as in Kutulakos's approach. Following that, however, instead of using the tangent points as an affine basis to determine stationary and non-stationary points, a correlational method based on the Hausdorff distance [13] is used to determine the stationarity of contour points. This is done by finding the best transformation aligning *virtual markings* from different frames. These *virtual markings* are made of artificial lines segments connecting points along the camera motion, and thus are stationary. Each real image is matched against the virtual marking image of the previous image in the sequence, using a line feature-based multidimensional Hausdorff distance pattern recognition algorithm [12]. The result of this process is a translation and scaling transformation between the virtual marking and points on the real image. Since markings are stationary, they will have the same translation and scaling property that the stationary points have. However, points on the occluding contour are not stationary and will have a different transformation. Therefore, by detecting the translational difference between consecutive images (while the scaling and translation trans-

*This work was supported in part by NSF grant IRI9309100 and in part by Penn State University and HRB Systems through the Center for Intelligent Information Processing.

1063-6919/97 $10.00 © 1997 IEEE

formations are applied to one of them), occluding contours can be distinguished from markings.

Our new approach has the following advantages: there is no need to search for and track contour correspondences between frames in contrast with all previous approaches; unlike the affine-based methods, the detection algorithm is robust to the occlusion of tangent points and appearance of new tangent points in following frames; it not only works for orthographic projection but also works with weak perspective; unlike Kutulakos's method; and finally, the detection algorithm is able to construct the full Euclidean surface structure by tracking the scaling from the consecutive images.

# 2 Active view point control

We actively control the viewpoint to achieve the best detection performance. Weak perspective projection is assumed in our approach. The camera motion is controlled such that there are as many points with tangents along the epipolar plane as possible. These points are special stationary points with respect to the epipolar plane, and will be used later for the detection.

The epipolar plane or camera motion plane correlates a set of special stationary points – the set of points with tangents along the same plane as the view motion plane. For example, suppose $P$ is a point on a visible rim or a marking curve of an object for a given viewpoint as shown in Figure 1. If the viewpoint is moved on $P$'s tangent plane, $P$ will remain on the visible rim or marking curve for as long as it does not become occluded by points that are closer to the camera. In this case, $P$ is a *stationary* point with respect to the viewpoint motion. However, if the camera does not move on $P$'s tangent plane, $P$ will not remain on the visible rim if $P$ was on a visible rim, while it will remain on the marking curve if it was on a marking curve. In the first case $P$ is a *non-stationary* point, but in the second case $P$ is a *stationary point.*

Even though surface marking points are stationary with respect to any camera motion plane, they do not provide any prior knowledge for their detection. On the other hand, the special stationary points described above are easy to detect, and can be used to distinguish occluding contours from markings.

# 3 Occluding contours detection

The shape and position of the visible rim depends on the shape of the surface and the viewpoint. This is the fundamental difference between visible curves and surface markings which is responsible for their different stationarity properties when the viewpoint is changed. We use the definition of stationarity for a surface curve stated by Kutulakos in [10]:

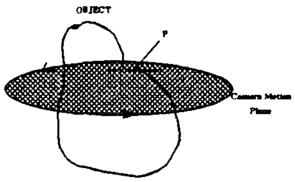


Figure 1: Illustration of stationarity with respect to camera motion plane

**Definition 3.1** *A surface curve is stationary if its position on the surface does not change when the viewpoint changes. It is non-stationary if its position is viewpoint-dependent.*

Following this definition, we can see that occluding contours are non-stationary while surface markings are stationary. Therefore, by matching stationary points between consecutive image frames, we can discriminate the non-stationary points of an occluding contour from those on a marking. Worthy of mention is that, there maybe a very few stationary points on occluding contours, but this does not affect the occluding contour classification, because most of non-stationary points on the occluding contour will not match.

## 3.1 Virtual Marking Construction

First, we detect all the points with tangents along the motion plane. These points are the special stationary points which are easy to pick, by looking at their gradient orientation. Then, we connect them with each other to construct a line pattern. We call this line pattern a *virtual marking image* (VMI). If $n$ points are selected, we can generate up to $\frac{1}{2}n(n-1)$ lines for the VMI. In order to improve the system's robustness, very short lines and lines enclosing triangles with small area are discarded. VMIs from two consecutive images are matched using the line feature-based Hausdorff distance method summarized below to find a translation, rotation, and scaling transformation between the frames. This transformation should put in agreement stationary points, such as markings, between the frames. However, non-stationary points on the occluding contours will not agree after being transformed.

## 3.2 Using the Hausdorff Distance

The Hausdorff distance is a measurement of the similarity of point sets and it can be used for pattern matching and tracking.

Given two point sets $A$ and $B$, the partial bidirectional Hausdorff distance between $A$ and $B$ is defined as [8]

$$H^{LK}(A, B) = max(h^L(A, B), h^K(B, A)). \quad (1)$$

where

$$h^K(B, A) = K^{th}_{b \in B} \min_{a \in A} \|a - b\| \quad (2)$$

and $\|.\|$ denotes the Euclidean distance. A similar definition can be given for $h^L(B, A)$. This partial distance measures the difference between portions of the two sets, making it more robust.

If a transformation $t(.)$ is applied to a model point set, we are interested in the Hausdorff distance as a function of the transformation of the set B, or $d(t) = H^{LK}(A, t(B))$. Therefore, to recognize a model-like image, we can apply different transformations to B; then, if $d(t)$ is less than a certain threshold, a match between the model and image can be hypothesized.

If $A$ is a set of images points, $B$ is a set of model points, the matching process of finding $d(t) = H^{LK}(A, t(B))$ is called forward matching. Given a threshold for the Hausdorff distance, the fraction of model points $f(t)$ with Hausdorff distance below this threshold is called the forward matching fraction. On the other hand, if $A$ is transformed instead of $B$, it is called the reverse matching process. Accordingly, given a reverse Hausdorff distance threshold there is a reverse matching fraction $g(t)$.

We have successfully developed a line feature based recognition (tracking) system [13, 12] to deal with rotation, scaling, and translation transformations between $A$ and $B$. In our approach, the prototype and the image can have different sizes and orientations. This is a common case when the camera moves around the object at varying distance or when the camera is rotated. The approach uses line segments extracted from the prototype and the image. Every line is represented in a four dimensional space by its middle point coordinates $x_m$ and $y_m$, the logarithm of its length $\log l$, and its orientation angle $\theta$. Then, the optimal four dimensional translation $t = (\nabla x, \nabla y, \nabla logl, \nabla\theta)$ corresponding to the translation, scaling, and rotation, respectively, is found by minimizing the Hausdorff distance between the prototype and the image. The details of the algorithms required to deal with the extremely large search space in an efficient manner are given in [12]. The method was tested systematically with synthetic data and real images. It was shown that it is a-robust and efficient algorithm, and that it can be applied to higher dimensional problems as well.

### 3.3 Determining Stationarity

From the previous section, it is seen that by applying the multidimensional pattern recognition algorithm [12] between the two VMIs, it is possible to find their translation, rotation, and scaling transformations. These transformations are first applied to the model image (the consecutive image); the transformed image is then templated on top of the image from the previous camera location (the first image). Let $\tau$ be the closest distance a point from one image can find for its matching point on the transformed consecutive image along the epipolar plane; $\tau$ is also called the displacement of that point. Let $D_{th}$ a predefined displacement threshold. Since all the markings should match well with each other, while the occluding contours should not, we use the following criteria to determine the stationarity of a point:

**Criteria 3.1** *A point is defined as an stationary point, if $\tau \leq D_{th}$ otherwise, it is a nonstationary point, where $D_{th}$ is a distance threshold.*

A contour is detected as an occluding contour if the ratio of the number of nonstationary points to stationary points is close to 1, and is detected as a marking contour if this ratio is close to 0.

# 4 Minimum camera movement

From the previous section, we know that while an image is overlapped with its consecutive transformed image, the displacement $\tau$ between the corresponding points defines whether they belong to a marking or an occluding contour. In the criteria 3.1, the displacement $\tau$ for the occluding contour is constrained by how much the camera view angle is changed and by the surface normal curvature. If the angle is too small, we may not observe enough displacement for the occluding contour, and misclassify it as marking. Also, the camera viewing angle for a small normal curvature curve needs not to change as much as that for a bigger normal curvature curve.

In the sequel, we develop a mathematical relation between the surface normal curvature and the required minimum camera view angle change for different distance thresholds for occluding contour classification. We assume that the camera moves on one epipolar plane. The result presented here applies to perspective projection model as well as weak perspective projection.

Figure 2 shows and object and two camera locations on one epipolar plane. At the first camera location, $P$ lies on the occluding contour, and $P'$ is the image of $P$. The second camera viewing direction $v_1$ has an

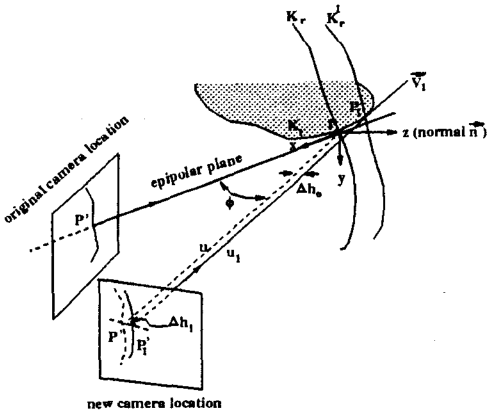


Figure 2: Illustration of camera view changing and the displacement of the occluding contours

angle of $\phi$ with the original viewing direction on the epipolar plane. At this new location, $P_1$ is on the occluding contour and its image is point $P_1'$. The point $P$, however, is no longer on the occluding contour. As shown in the Figure, let $P$ be the origin of the Cartesian coordinate system $xyz$ where $z$ is along the normal direction and $x$ is along the opposite direction to the first camera viewing direction. Therefore, the $x - z$ plane is along the normal plane or epipolar plane and $x - y$ is on the tangent plane to the object at point $P$. The normal plane meets the object on the normal curve with normal curvature $K_t$. The radial curvatures of the occluding contours at $P$ and $P_1$ are denoted as $K_r$ and $K_r^1$, respectively. The surface near $P$ can be described as:

$$z(x,y) = \frac{1}{2}(ax^2 + abxy + cy^2) + O(3)$$

By neglecting the third and higher order terms, Koenderink [9] showed that the normal curve can be described as $(y = 0)$:

$$z = \frac{1}{2}ax^2 = \frac{1}{2}K_t x^2 \tag{3}$$

Then, the tangents of this curve are defined by:

$$\frac{\partial z}{\partial x} = K_t x \tag{4}$$

We now focus on obtaining the distance between $P$ and $P_1$ in the direction perpendicular to the view line $v_1$. We denote this distance as $\Delta h_o$. In fact, the image $\Delta h_i$ of $\Delta h_o$ is what can be computed with the proposed correlational method - i.e. the displacement $\tau$ is $\Delta h_i$.

Since the second line of sight $v_1$ is on the $x - z$ plane and has an angle of $\phi$ with respect to the $x$ axis, its slope is $\tan\phi$. Since $v_1$ is tangent to the normal curve at $P_1$, the coordinates of $P_1$ on the $x - z$ plane can be found by equating Equation 4 with $\tan\phi$ and using Equation 3. As a result, $P$'s coordinates are $(\frac{\tan\phi}{K_t}, \frac{1}{2K_t}\tan^2\phi)$.

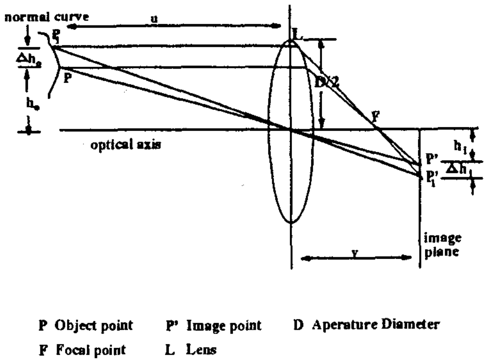


Figure 3: Geometric optics for the normal curve on the epipolar plane

The line equation for $v_1$ is then:

$$z - x\tan\phi + \frac{\tan^2\phi}{2K_t} = 0 \tag{5}$$

Since $P$ has coordinates $(0,0)$ on the $x - z$ plane, the perpendicular distance between $P$ and the line of sight $v_1$ is obtained as:

$$\Delta h_o = \frac{\frac{\tan^2\phi}{2K_t}}{\sqrt{1 + \tan^2\phi}} = \frac{\cos\phi\tan^2\phi}{2K_t} \tag{6}$$

In the case of orthographic projection, $\Delta h_o$ is equal to $\Delta h_i$ or $\tau$. We now compute $\Delta h_i$ in the case of perspective projection.

The image formation on the epipolar plane is illustrated by Figure 3. Suppose that the distance between $P$ and the lens is $u$, the focal length is $f$, and the distance between the and the lens is $v$. The height of $P$ is $h_o$, and the height of its image is $h_i$. From the lens equation and triangulation[6], we obtain:

$$h_i = h_o\frac{f}{u - f} \tag{7}$$

The same formula applies to $P_1$ with its $u_1$, $h_o^1$, and $h_i^1$. For very small differences between the distances between the the lens and the points $P$ and $P_1$, the defocusing effect can be ignored.

By using Equation 7, the image of the height difference $\Delta h_o$ for these two points in the direction perpendicular to the optical axis is:

$$\Delta h_i = h_o^1\frac{f}{u_1 - f} - h_o\frac{f}{u - f} \approx \Delta h_o\frac{f}{u - f} \tag{8}$$

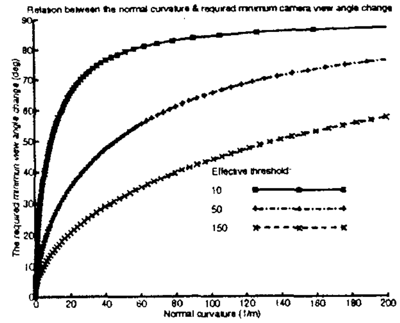


Figure 4: Plot of the required minimum camera viewing angle change vs. normal curvature

Substituting Equation 6 into the above equation, we obtain the final expression for $h_i$:

$$\Delta h_i = \frac{f}{2(u-f)} \frac{\cos\phi \tan^2\phi}{K_t} \quad (9)$$

In fact, in the case of weak perspective projection, this distance is what we can compute with the Hausdorff distance matching approach, the result is the displacement $\tau$. By setting $\tau$ equal to $\Delta h_i$, from the above equation, we have:

$$K_t = \frac{f}{2(u-f)} \frac{1}{\tau} \cos\phi \tan^2\phi \quad (10)$$

If we set $\tau$ equal to the predefined distance threshold $D_{th}$, we can find the minimum angle $\phi$ to distinguish occluding contours. We set an effective threshold $K_{eff}$ as $\frac{f}{2(u-f)} \frac{1}{D_{th}}$, then obtain the relation between the minimum view angle change, the normal curvature, and the effective threshold:

$$K_t = K_{eff} \cos\phi \tan^2\phi \quad (11)$$

The result is plotted in Figure 4. From the plot, it is seen that the normal curve with smaller curvature requires smaller view angle change for the occluding contour detection. Also, the bigger the effective threshold is or the smaller the distance threshold is, the larger view angle change is required. This is a very helpful observation, as illustrated in the experiments with real images, since it can be used to plan for the next viewing direction.

# 5 Experimental results

The detection algorithm was implemented with C/C++. Simulations and real image tests were performed to characterize the algorithm properties.

## 5.1 Simulation results

The coordinates of the tangent points are the very first input to the algorithm. This input affects the scaling and translation computation for the consecutive images. As a result it will affect the occluding contour and marking detection when overlapping is performed. Furthermore, noise in the image could vary the coordinates of the tangent points. Finally, some tangent points can be occluded and new tangent points can appear when the camera is moved to a new location. Therefore, we should explore the statistics of correctly computing the scaling and translation versus noise model under different percentages of occluded tangent points.

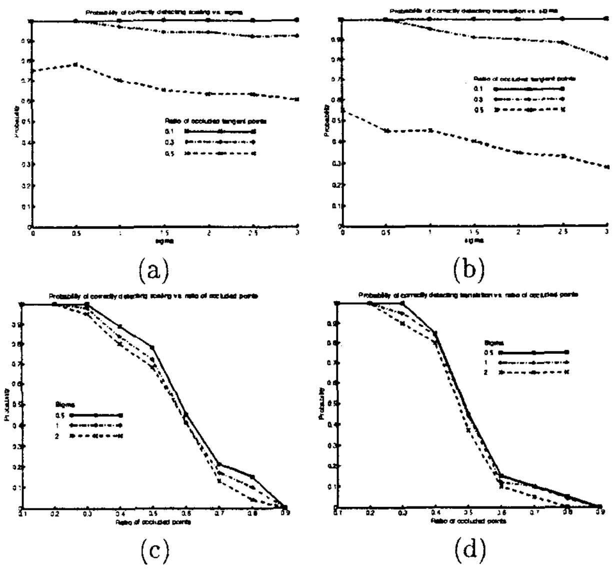
We simulate this process by using 512 by 512 images. Twenty points are uniformly selected with their coordinates ranging between 0 and 512. These points are assumed to be the detected tangent points from the image taken from the camera's first viewing direction. They are then scaled and translated into a new set of coordinates which correspond to the detected tangent points from a new viewing direction. The coordinates of this new set are perturbed using a Gaussian distribution with zero mean and standard deviation of $\sigma$. To simulate occlusion, we randomly remove some points from the transformed set of points. The same number of new points are also added to this set to simulate the possible appearance of new tangent points. Therefore, the total number of tangent points from two consecutive images is assumed to be the same.

For each $\sigma$ and ratio of occluded points, the system is tested 20 times. The obtained translation and scaling transformations are then compared with the ground truth to obtain the following probabilities:

- The probability of correctly obtaining the scaling vs. $\sigma$.
- The probability of correctly obtaining the translation vs. $\sigma$.
- The probability of correctly obtaining the scaling vs. the ratio of occluded tangent points.
- The probability of correctly obtaining the translation vs. the ratio of occluded tangent points.

Where, for correctly computing the scaling, we mean the difference between the obtained scaling and the ground truth is as small as 0.1, and for correctly computing the translation, we mean the square root of the sum of the squared coordinate differences between the obtained values and the ground truth is as small as 2.

Figures 5 (a) and (b) show that the probabilities of correctly detecting scaling and translation are not very sensitive to the change of $\sigma$. Before the ratio of



(a) (b)

(c) (d)

Figure 5: The simulation results for the probabilities

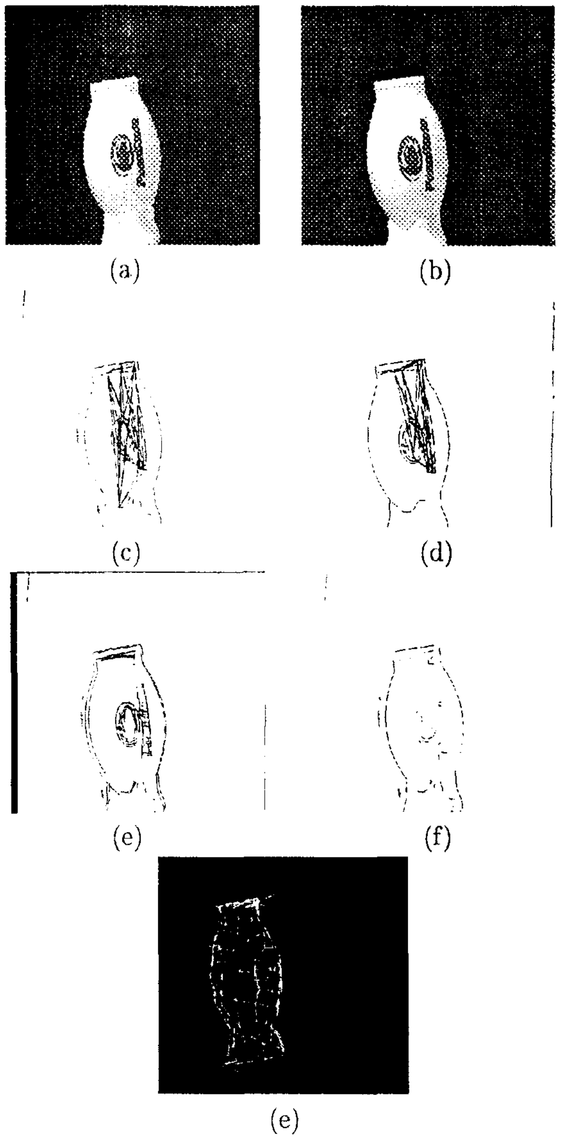
the occluded tangent points reaches 40% to 50%, the probabilities of correctly detecting scaling and translation is quite robust and higher than 0.8 (see Figures 5 (c) and (d)). However, with the further increase of this ratio, these probabilities decrease dramatically.

## 5.2 Real image tests

In this section we present an example with real images. These images are taken 5 degrees apart on a horizontal plane. Virtual markings are formed by connecting points with horizontal tangents with lines that are at least 30 pixels long and such that form triangles enclosing at least 350 pixels.

In this example, we focus on obtaining the probability of a point belonging to a marking or occluding contour with respect to different distance thresholds. The histogram of the number of points on a curve with different possible displacements found through templating is a good measurement of this probability. This test can verify the correctness of our detection algorithm.

Figure 6 shows the 3rd and fourth frames of a football. Points with horizontal tangents were selected from each image to construct their VMIs. The VMIs are shown in (c) and (d) which are overlapped on the corresponding edge detected images. The line feature-based Hausdorff distance matching algorithm was then applied to the VMIs. The obtained results show that, the first image of the football is 0.98 smaller than the second image. The translation for the football were found to be (13, 1). The transformed second frame is overlaid on top of the first frame in Figure 6(e). It is seen that markings matched very well. On the other hand, there is a significant displacement between the occluding contours. By choosing $D_{th}$ as 2, we obtain the final reconstructed occluding contour image of the football as shown in Figure 6(f). The noise on the reconstructed contours can be removed first by performing the reverse matching process, or in this case, by matching the third frame with the second frame, and then performing a logical AND between the resultant reconstructed image and Figure 6(f).



(a) (b)

(c) (d)

(e) (f)

(e)

Figure 6: Football example. (a) and (b) are two frames, (c) and (d) are the VMIs and edge detected images, (e) is the matching result, (f) is the final detected contour, and (g) shows the final optimized meshed football.

The histogram for the football is studied as shown

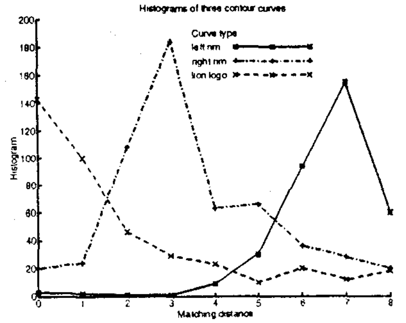


Figure 7: Histograms for three contour curves on the football

in Figure 7. The histograms for the left rim, right rim, and the "PennState" lion logo are plotted in terms of found matching distance from the consecutive images. It is seen that, there exists a good threshold (eg. 4) to separate the left rim from the logo. On the other hand, the right rim and the logo do not separate from each other very well. A threshold of 2 will separate most of the points of the right rim from the points of the logo. One more important observation, from the analysis in the previous section and Figure 4, is that the left rim has smaller normal curvatures than the right rim because it does not give as much displacement or matching distance as the left rim does for the same viewing angle. As a result, 5 degrees of camera viewing angle change is a good choice for distinguishing the left rim from the logo, but not for distinguishing the right rim from the logo.

After obtaining a sequence of 2D contour data, we first converted them back into 3D space using Zheng's method [14], then used Hoppe's mesh optimization method [7] to obtain the final 3D representation of the object shown shown in Figure 6(g).

# 6 Conclusion

In this paper, a new method for detecting occluding contours from markings for 3D object modeling is proposed. Instead of using a computational method or an affine transformation to determine stationary and non-stationary points, we use a correlational method based on the Hausdorff distance. This method does not require prior known markings, nor to search and track feature point correspondences between frames, nor to accurately calibrate the camera, nor to know the camera motion. It works well under weak perspective projection and it is robust to the occlusion of the tangent points between consecutive images. We also presented a theoretical expression for the minimum required viewing angle change for certain surface normal curvature which provides a qualitative relation between the camera motion and the surface structure.

# References

[1] E. Arbogast and R. Mohr. 3d structure inference from image sequences. *Journal of Pattern Recognition and Artificial Intelligence*, 5(5), 1991.

[2] R. Cipolla and A. Blake. Qualitative surface shape from deformation of image curves. *International Journal of Computer Vision*, 8(1):53–69, 1992.

[3] R. Cipolla and A. Blake. Surface shape from deformation of apparent contours. *International Journal of Computer Vision*, 9(2):83–112, 1992.

[4] P. Giblin and R. Weiss. Epipolar curves on surfaces. *Image and Vision Computing*, 13:33–44, 1987.

[5] P. Giblin and R. Weiss. Reconstruction of surfaces from profiles. *Proc. 1st Int. Conf. on Computer Vision*, pages 136–144, 1987.

[6] E. Hecht. *Optics*. Addison-Wesley Publishing, 1990.

[7] H. Hoppe, T. DeRose, T. Duchamp, J. McDonald, and W. Stuetzle. Surface reconstruction from unorganized points. *Computer Graphics(SIGGRAPH'92 Proceedings)*, 26(2):71–78, 1992.

[8] D. P. Huttenlocher, G. Klanderman, and W. Rucklidge. Comparing images using the Hausdorff distance. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 15:850–863, September 1993.

[9] J. J. Koenderink. What does the occluding contour tell us about solid shape? *Perception*, 13, 1984.

[10] K. N. Kutulakos. Exploring three-dimensional objects by controlling the point of observation. *PhD thesis, University of Wisconsin-Madison*, 1994.

[11] R. Vaillant and O. D. Faugeras. Using extremal boundaries for 3d object modeling. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 14(2):157–153, 1992.

[12] X. Yi. *3D Object Reconstruction from Gray Scale Images Using Active Vision.* PhD thesis, Dept. of Electrical Engineering, The Pennsylvania State University, University Park, PA, 16802, 1997.

[13] X. Yi and O. Camps. Line feature-based recognition using Hausdorff distance. *IEEE Int. Symposium on Computer Vision, November 21-23*, pages 79–84, 1995.

[14] J. Y. Zheng. Acquiring 3-d models from sequences of contours. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 16(2):163–178, 1994.

[15] A. Zisserman, A. Blake, C. A. Rothwell, L. J. van Gool, and M. V. Diest. Eliciting qualitative structure from image curve deformations. *Proc. 4th Int. Conf. on Computer Vision*, pages 340–35, 1993.